



# Adobe FrameMaker

## Scripting Guide



September 2022

---

# Contents

- Overview . . . . . 1**
  - About this document . . . . . 1
    - Naming conventions . . . . . 1
    - Typographic conventions . . . . . 2
  - Scripting support in FrameMaker . . . . . 2
    - Development and debugging tools in ExtendScript toolkit . . . . . 2
    - ExtendScript capabilities . . . . . 2
  - Creating Scripts . . . . . 3
  - Accessing ESTK . . . . . 3
    - Connect ESTK to FrameMaker . . . . . 3
    - To write, edit, and run scripts . . . . . 4
  - Using script library . . . . . 4
  - Using object model viewer . . . . . 5
  - ExtendScript use cases . . . . . 5
  
- What's new in ESTK . . . . . 7**
  - Export and import XLIFF . . . . . 7
  
- Sample ESTK scripts . . . . . 8**
  - Adding text to a document and enabling change bars . . . . . 8
  - Changing the table properties in a document . . . . . 8
  
- Differences between scripts and FDK . . . . . 10**
  - Naming differences between scripts and FDK . . . . . 10
  - Notifications . . . . . 13
  - Menus and Commands . . . . . 15

---

<b>Object Reference</b> . . . . .	<b>17</b>
AFrame . . . . .	17
AFrame methods . . . . .	24
Arc . . . . .	24
Arc methods . . . . .	28
AttrCondExpr . . . . .	29
AttrCondExpr methods . . . . .	29
Attribute . . . . .	29
Attribute methods . . . . .	29
AttributeDef . . . . .	30
AttributeDef methods . . . . .	32
AttributeDefs . . . . .	32
AttributeDefs methods . . . . .	32
AttributeEx . . . . .	32
Attributes . . . . .	32
Attributes methods . . . . .	33
AttributesEx . . . . .	33
AttributesEx methods . . . . .	33
BodyPage . . . . .	33
BodyPage methods . . . . .	35
Book . . . . .	35
Book methods . . . . .	48
BookComponent . . . . .	49
BookComponent methods . . . . .	66
Cell . . . . .	67
Cell methods . . . . .	70
CharFmt . . . . .	70
CharFmt methods . . . . .	75
CMSCheckinParam . . . . .	76
CMSDeleteParam . . . . .	76
CMSInfo . . . . .	77
CMSMenuItem . . . . .	77
CMSProperty . . . . .	78
CMSProperties . . . . .	78
CMSRegistration . . . . .	78
CMSResult . . . . .	79
CMSSession . . . . .	79
Color . . . . .	79
Color methods . . . . .	81
Column . . . . .	81
Column methods . . . . .	82
CombinedFont . . . . .	82
CombinedFont methods . . . . .	82
CombinedFontDefn . . . . .	83
CombinedFontDefn methods . . . . .	83
CombinedFonts . . . . .	84

---

CombinedFonts methods . . . . .	.84
Command . . . . .	.84
Command methods . . . . .	.91
CompareRet . . . . .	.92
CompareRet methods . . . . .	.92
CondFmt . . . . .	.92
CondFmt methods . . . . .	.94
Constants . . . . .	.94
Doc . . . . .	295
Doc methods . . . . .	345
Element . . . . .	346
Element methods . . . . .	354
ElementCatalogEntries . . . . .	354
ElementCatalogEntries methods . . . . .	354
ElementCatalogEntry . . . . .	355
ElementCatalogEntry methods . . . . .	355
ElementDef . . . . .	355
ElementDef methods . . . . .	358
ElementLoc . . . . .	358
ElementLoc methods . . . . .	359
ElementRange . . . . .	359
ElementRange methods . . . . .	359
Ellipse . . . . .	359
Ellipse methods . . . . .	363
FCodes . . . . .	363
FilterArgs . . . . .	429
FilterArgs methods . . . . .	429
Flow . . . . .	429
Flow methods . . . . .	430
FmtChangeList . . . . .	431
FmtChangeList methods . . . . .	439
FmtRule . . . . .	440
FmtRule methods . . . . .	440
FmtRuleClause . . . . .	441
FmtRuleClause methods . . . . .	442
Fn . . . . .	443
Fn methods . . . . .	444
Font . . . . .	444
Font methods . . . . .	444
Fonts . . . . .	444
Fonts methods . . . . .	444
Group . . . . .	444
Group methods . . . . .	448
HiddenPage . . . . .	448
HiddenPage methods . . . . .	449
GraphicsFmt . . . . .	449

---

InlineComponent . . . . .	450
InlineComponent methods . . . . .	451
Inset methods . . . . .	451
IdValuePair . . . . .	451
IdValuePair methods . . . . .	451
IdValuePairs . . . . .	451
Inset . . . . .	452
Inset methods . . . . .	458
Ints . . . . .	458
Ints methods . . . . .	458
KeyCatalog . . . . .	459
Line . . . . .	459
Line methods . . . . .	463
Marker . . . . .	464
Marker methods . . . . .	464
MarkerType . . . . .	464
MarkerType methods . . . . .	465
MasterPage . . . . .	465
MasterPage methods . . . . .	466
Math . . . . .	466
Math methods . . . . .	471
MathML . . . . .	471
MathML methods . . . . .	476
Menu . . . . .	476
Menu methods . . . . .	477
MenuItemSeparator . . . . .	477
MenuItemSeparator methods . . . . .	477
Metrics . . . . .	477
Metrics methods . . . . .	478
Pgf . . . . .	478
Pgf methods . . . . .	487
PgfFmt . . . . .	487
PgfFmt methods . . . . .	497
Point . . . . .	497
Point methods . . . . .	497
Points . . . . .	497
Points methods . . . . .	497
Polygon . . . . .	497
Polygon methods . . . . .	502
Polyline . . . . .	502
Polyline methods . . . . .	506
PropIdent . . . . .	507
PropIdent methods . . . . .	507
PropVal . . . . .	507
PropVal methods . . . . .	507
PropVals . . . . .	507

---

PropVals methods . . . . .	507
Rectangle . . . . .	508
Rectangle methods . . . . .	512
RefPage . . . . .	512
RefPage methods . . . . .	512
RoundRect . . . . .	513
RoundRect methods . . . . .	517
Row . . . . .	517
Row methods . . . . .	519
Rubi . . . . .	519
Rubi methods . . . . .	520
RulingFmt . . . . .	520
RulingFmt methods . . . . .	521
Strings . . . . .	521
Strings methods . . . . .	521
SubCol . . . . .	521
SubCol methods . . . . .	522
Tab . . . . .	522
Tab methods . . . . .	523
Tabs . . . . .	523
Tabs methods . . . . .	524
Tbl . . . . .	524
Tbl methods . . . . .	529
TblFmt . . . . .	529
TblFmt methods . . . . .	533
TextFrame . . . . .	534
TextFrame methods . . . . .	539
TextItem . . . . .	539
TextItems . . . . .	543
TextItems methods . . . . .	543
TextLine . . . . .	543
TextLine methods . . . . .	549
TextLoc . . . . .	549
TextLoc methods . . . . .	550
TextRange . . . . .	550
TextRange methods . . . . .	550
TiApiClient . . . . .	550
TiApiClient methods . . . . .	551
TiFlow . . . . .	552
TiFlow methods . . . . .	553
TiText . . . . .	553
TiText methods . . . . .	555
TiTextTable . . . . .	555
TiTextTable methods . . . . .	556
TypedVal . . . . .	557
TypedVals . . . . .	557

TypedVals methods . . . . .	558
UBytes . . . . .	558
UBytes methods . . . . .	558
UInts . . . . .	558
UInts methods . . . . .	558
UnanchoredFrame . . . . .	558
UnanchoredFrame methods . . . . .	563
Var . . . . .	563
Var methods . . . . .	564
VarFmt . . . . .	564
VarFmt methods . . . . .	566
XRef . . . . .	566
XRef methods . . . . .	567
XRefFmt . . . . .	568
XRefFmt methods . . . . .	568
app . . . . .	568
app methods . . . . .	577
<b>Function Summary . . . . .</b>	<b>578</b>
AFrame . . . . .	578
Delete . . . . .	578
Description . . . . .	578
Returns . . . . .	578
Syntax . . . . .	578
GetProps . . . . .	579
Description . . . . .	579
Returns . . . . .	579
Syntax . . . . .	579
SetProps . . . . .	579
Description . . . . .	579
Returns . . . . .	581
Syntax . . . . .	581
Parameters . . . . .	581
ObjectValid . . . . .	581
Description . . . . .	581
Returns . . . . .	582
Syntax . . . . .	582
Arc . . . . .	582
Delete . . . . .	582
Description . . . . .	582
Returns . . . . .	582
Syntax . . . . .	582
GetProps . . . . .	582
Description . . . . .	582
Returns . . . . .	582
Syntax . . . . .	582

---

SetProps . . . . .	583
Description . . . . .	583
Returns . . . . .	583
Syntax . . . . .	583
Parameters . . . . .	583
ObjectValid . . . . .	583
Description . . . . .	583
Returns . . . . .	583
Syntax . . . . .	583
AttrCondExpr . . . . .	583
ApplyAttributeExpression . . . . .	583
Description . . . . .	583
Returns . . . . .	584
Syntax . . . . .	584
GetProps . . . . .	584
Description . . . . .	584
Returns . . . . .	584
Syntax . . . . .	584
SetProps . . . . .	584
Description . . . . .	584
Returns . . . . .	584
Syntax . . . . .	584
Parameters . . . . .	584
ObjectValid . . . . .	585
Description . . . . .	585
Returns . . . . .	585
Syntax . . . . .	585
Attribute . . . . .	585
Attribute . . . . .	585
Description . . . . .	585
Returns . . . . .	585
Syntax . . . . .	585
Parameters . . . . .	585
AttributeDef . . . . .	586
AttributeDef . . . . .	586
Description . . . . .	586
Returns . . . . .	586
Syntax . . . . .	586
Parameters . . . . .	586
AttributeDefs . . . . .	587
concat . . . . .	587
Description . . . . .	587
Returns . . . . .	587
Syntax . . . . .	588
Parameters . . . . .	588
pop . . . . .	588



---

Description .....	588
Returns .....	588
Syntax .....	588
push .....	588
Description .....	588
Returns .....	588
Syntax .....	588
Parameters .....	588
Attributes .....	589
concat .....	589
Description .....	589
Returns .....	589
Syntax .....	589
Parameters .....	589
pop .....	589
Description .....	589
Returns .....	589
Syntax .....	589
push .....	589
Description .....	589
Returns .....	590
Syntax .....	590
Parameters .....	590
AttributesEx .....	590
concat .....	590
Description .....	590
Returns .....	590
Syntax .....	590
Parameters .....	590
pop .....	591
Description .....	591
Returns .....	591
Syntax .....	591
push .....	591
Description .....	591
Returns .....	591
Syntax .....	591
Parameters .....	591
BodyPage .....	591
Delete .....	591
Description .....	591
Returns .....	592
Syntax .....	592
GetProps .....	592
Description .....	592
Returns .....	592

---

Syntax	592
SetProps	592
Description	592
Returns	592
Syntax	592
Parameters	592
ObjectValid	593
Description	593
Returns	593
Syntax	593
ApplyPageLayout	593
Description	593
Returns	593
Syntax	593
Parameters	594
Book	594
Compare	594
Description	594
Returns	595
Syntax	595
Parameters	595
NewBookComponentInHierarchy	596
Description	596
Returns	596
Syntax	596
Parameters	597
SilentPrintDoc	597
Description	597
Returns	597
Syntax	597
SimpleGenerate	597
Description	597
Returns	598
Syntax	598
Parameters	598
SimpleImportFormats	598
Description	598
Returns	600
Syntax	600
Parameters	600
SimpleImportElementDefs	600
Description	600
Returns	601
Syntax	601
Parameters	601
GetUniqueObject	601

---

Description .....	601
Returns .....	602
Syntax .....	602
Parameters .....	602
Close .....	602
Description .....	602
Returns .....	602
Syntax .....	602
Parameters .....	603
SimpleSave .....	603
Description .....	603
Returns .....	604
Syntax .....	604
Parameters .....	605
NewSeriesObject .....	605
Description .....	605
Returns .....	605
Syntax .....	606
Parameters .....	606
NewSeriesBookComponent .....	606
Description .....	606
Returns .....	606
Syntax .....	606
Parameters .....	606
NewBookComponentOfTypeInHierarchy .....	607
Description .....	607
Returns .....	607
Syntax .....	607
Parameters .....	607
GetConditionalSettings .....	607
Description .....	607
Returns .....	607
Syntax .....	607
ApplyConditionalSettings .....	607
Description .....	607
Returns .....	608
Syntax .....	608
Parameters .....	608
ManageConditionalExpressions .....	609
Description .....	609
Returns .....	609
Syntax .....	609
Parameters .....	609
GetConditionalExpression .....	610
Description .....	610
Returns .....	610

---

Syntax	610
Parameters	610
Save	611
Description	611
Returns	612
Syntax	612
Parameters	613
UpdateBook	613
Description	613
Returns	617
Syntax	617
Parameters	617
Import	617
Description	617
Returns	630
Syntax	631
Parameters	631
GetProps	631
Description	631
Returns	631
Syntax	631
SetProps	631
Description	631
Returns	632
Syntax	632
Parameters	632
ObjectValid	632
Description	632
Returns	632
Syntax	632
BookComponent	632
Delete	632
Description	632
Returns	632
Syntax	633
MoveComponent	633
Description	633
Returns	633
Syntax	633
Parameters	633
GetProps	633
Description	633
Returns	633
Syntax	633
SetProps	633
Description	633

---

Returns	634
Syntax	634
Parameters	634
ObjectValid	634
Description	634
Returns	634
Syntax	634
Cell	634
StraddleCells	634
Description	634
Returns	635
Syntax	635
Parameters	635
UnStraddleCells	635
Description	635
Returns	636
Syntax	636
Parameters	636
GetText	636
Description	636
Returns	641
Syntax	641
Parameters	641
GetProps	641
Description	641
Returns	642
Syntax	642
SetProps	642
Description	642
Returns	642
Syntax	642
Parameters	642
ObjectValid	642
Description	642
Returns	642
Syntax	642
CharFmt	643
Delete	643
Description	643
Returns	643
Syntax	643
GetProps	643
Description	643
Returns	643
Syntax	643
SetProps	643

---

Description .....	643
Returns .....	643
Syntax .....	643
Parameters .....	644
ObjectValid .....	644
Description .....	644
Returns .....	644
Syntax .....	644
CMXObject .....	644
CMS SetProperty() .....	644
Description .....	644
Returns .....	644
Syntax .....	644
Parameters .....	645
CMS Set Properties() .....	645
Description .....	645
Returns .....	645
Syntax .....	645
Parameters .....	646
CMS Get Property() .....	646
Description .....	646
Returns .....	646
Syntax .....	647
Parameters .....	647
CMS Get Properties() .....	647
Description .....	647
Returns .....	647
Syntax .....	647
Parameters .....	648
CMS Enable Command() .....	648
Description .....	648
Returns .....	648
Syntax .....	648
Parameters .....	648
CMS Disable Command() .....	649
Description .....	649
Returns .....	649
Syntax .....	649
Parameters .....	649
CMS Show Checkout UI() .....	650
Description .....	650
Returns .....	650
Syntax .....	650
Parameters .....	650
CMS Show Checkin UI() .....	651
Description .....	651

---

Returns	651
Syntax	651
Parameters	651
CMSShowCancelCheckoutUI()	652
Description	652
Returns	652
Syntax	652
Parameters	652
CMSShowDeleteUI()	653
Description	653
Returns	653
Syntax	653
Parameters	653
CMSShowCommonListUI()	654
Description	654
Returns	654
Syntax	654
Parameters	654
CMSShowPropertyUI()	655
Description	655
Returns	655
Syntax	655
Parameters	655
CMSShowPropertyUIWithTitle	656
Description	656
Returns	656
Syntax	656
Parameters	656
CMSRegistration	657
CMSGetCMSInfo()	657
Description	657
Returns	657
Syntax	657
Parameters	657
CMSConfigLoginUI()	657
Description	657
Returns	657
Syntax	658
Parameters	658
CMSSession	658
CMSLogout()	658
Description	658
Returns	658
Syntax	658
Parameters	658
GetCMSObjectFromPath()	659

---

Description .....	659
Returns .....	659
Syntax .....	659
Parameters .....	659
CMSShowBrowseRepositoryUI .....	659
Description .....	659
Returns .....	659
Syntax .....	660
Parameters .....	660
CMSCreateObject .....	660
Description .....	660
Returns .....	660
Syntax .....	660
Parameters .....	660
CMSSetCmsIdFromSession .....	661
Description .....	661
Returns .....	661
Syntax .....	661
Parameters .....	661
Color .....	661
Delete .....	661
Description .....	661
Returns .....	661
Syntax .....	661
GetProps .....	662
Description .....	662
Returns .....	662
Syntax .....	662
SetProps .....	662
Description .....	662
Returns .....	662
Syntax .....	662
Parameters .....	662
ObjectValid .....	662
Description .....	662
Returns .....	663
Syntax .....	663
CombinedFont .....	663
CombinedFont .....	663
Description .....	663
Returns .....	663
Syntax .....	663
Parameters .....	663
CombinedFontDefn .....	664
CombinedFamilyFonts .....	664
Description .....	664



---

Returns	664
Syntax	664
Delete	664
Description	664
Returns	664
Syntax	664
GetProps	664
Description	664
Returns	664
Syntax	665
SetProps	665
Description	665
Returns	665
Syntax	665
Parameters	665
ObjectValid	665
Description	665
Returns	665
Syntax	665
CombinedFonts	666
concat	666
Description	666
Returns	666
Syntax	666
Parameters	666
pop	666
Description	666
Returns	666
Syntax	666
push	666
Description	666
Returns	667
Syntax	667
Parameters	667
Command	667
Delete	667
Description	667
Returns	667
Syntax	667
GetProps	667
Description	667
Returns	667
Syntax	668
SetProps	668
Description	668
Returns	668

---

Syntax	668
Parameters	668
ObjectValid	668
Description	668
Returns	668
Syntax	668
Delete	669
Description	669
Returns	669
Syntax	669
CompareRet	669
CompareRet	669
Description	669
Returns	669
Syntax	669
Parameters	669
CondFmt	670
Delete	670
Description	670
Returns	670
Syntax	670
DeleteCondTag	670
Description	670
Returns	670
Syntax	670
Parameters	670
GetProps	671
Description	671
Returns	671
Syntax	671
SetProps	671
Description	671
Returns	671
Syntax	671
Parameters	672
ObjectValid	672
Description	672
Returns	672
Syntax	672
Doc	672
AddText	672
Description	672
Returns	673
Syntax	673
Parameters	673
CenterOnText	673

---

Description .....	673
Returns .....	674
Syntax .....	674
Parameters .....	674
Clear .....	674
Description .....	674
Returns .....	675
Syntax .....	675
Parameters .....	675
ClearAllChangebars .....	676
Description .....	676
Returns .....	676
Syntax .....	676
Compare .....	676
Description .....	676
Returns .....	677
Syntax .....	677
Parameters .....	677
Copy .....	678
Description .....	678
Returns .....	679
Syntax .....	679
Parameters .....	679
Cut .....	679
Description .....	679
Returns .....	680
Syntax .....	680
Parameters .....	680
DeleteText .....	681
Description .....	681
Returns .....	681
Syntax .....	681
Parameters .....	681
DemoteElement .....	681
Description .....	681
Returns .....	682
Syntax .....	682
GetTextForRange .....	682
Description .....	682
Returns .....	687
Syntax .....	687
Parameters .....	687
GetTextForRange2 .....	687
Description .....	687
Returns .....	688
Syntax .....	688

---

Parameters .....	688
GetTextProps .....	688
Description .....	688
Returns .....	689
Syntax .....	689
Parameters .....	689
GetTextPropVal .....	689
Description .....	689
Returns .....	690
Syntax .....	690
Parameters .....	690
GetTextVal .....	690
Description .....	690
Returns .....	691
Syntax .....	691
Parameters .....	691
HypertextCommand .....	692
Description .....	692
Returns .....	692
Syntax .....	692
Parameters .....	692
MergeIntoFirst .....	692
Description .....	692
Returns .....	693
Syntax .....	693
MergeIntoLast .....	693
Description .....	693
Returns .....	693
Syntax .....	694
NewTable .....	694
Description .....	694
Returns .....	695
Syntax .....	695
Parameters .....	695
Paste .....	696
Description .....	696
Returns .....	697
Syntax .....	697
Parameters .....	697
PromoteElement .....	697
Description .....	697
Returns .....	698
Syntax .....	698
QuickSelect .....	698
Description .....	698
Returns .....	699

---

Syntax	699
Parameters	699
Redisplay	699
Description	699
Returns	699
Syntax	699
Reformat	700
Description	700
Returns	700
Syntax	700
Rehyphenate	700
Description	700
Returns	700
Syntax	701
ResetEqnSettings	701
Description	701
Returns	701
Syntax	701
ResetReferenceFrames	701
Description	701
Returns	702
Syntax	702
RestartPgfNumbering	702
Description	702
Returns	702
Syntax	702
ScrollToText	702
Description	702
Returns	703
Syntax	703
Parameters	703
SetTextProps	703
Description	703
Returns	704
Syntax	704
Parameters	705
SetTextPropVal	705
Description	705
Returns	706
Syntax	706
Parameters	706
SetTextVal	706
Description	706
Returns	707
Syntax	708
Parameters	708

---

SilentPrintDoc . . . . .	708
Description . . . . .	708
Returns . . . . .	709
Syntax . . . . .	709
SimpleImportFormats . . . . .	709
Description . . . . .	709
Returns . . . . .	710
Syntax . . . . .	710
Parameters . . . . .	710
SplitElement . . . . .	711
Description . . . . .	711
Returns . . . . .	712
Syntax . . . . .	712
TextLocToElementLoc . . . . .	712
Description . . . . .	712
Returns . . . . .	712
Syntax . . . . .	712
Parameters . . . . .	712
UnWrapElement . . . . .	713
Description . . . . .	713
Returns . . . . .	713
Syntax . . . . .	713
UpdateXRefs . . . . .	713
Description . . . . .	713
Returns . . . . .	714
Syntax . . . . .	714
Parameters . . . . .	714
UpdateDITAResferences . . . . .	714
Description . . . . .	714
Returns . . . . .	715
Syntax . . . . .	715
Parameters . . . . .	715
SimpleImportElementDefs . . . . .	715
Description . . . . .	715
Returns . . . . .	716
Syntax . . . . .	716
Parameters . . . . .	716
GetUniqueObject . . . . .	717
Description . . . . .	717
Returns . . . . .	717
Syntax . . . . .	717
Parameters . . . . .	717
Close . . . . .	717
Description . . . . .	717
Returns . . . . .	718
Syntax . . . . .	718

---

Parameters .....	718
SimpleSave .....	718
Description .....	718
Returns .....	719
Syntax .....	720
Parameters .....	720
NewGraphicObject .....	720
Description .....	720
Returns .....	721
Syntax .....	721
Parameters .....	721
NewArc .....	721
Description .....	721
Returns .....	721
Syntax .....	722
Parameters .....	722
NewEllipse .....	722
Description .....	722
Returns .....	722
Syntax .....	722
Parameters .....	722
NewFlow .....	723
Description .....	723
Returns .....	723
Syntax .....	723
Parameters .....	723
NewGroup .....	723
Description .....	723
Returns .....	723
Syntax .....	723
Parameters .....	724
NewInlineComponentOfType .....	724
Description .....	724
Returns .....	724
Syntax .....	724
Parameters .....	724
NewInset .....	725
Description .....	725
Returns .....	725
Syntax .....	725
Parameters .....	725
NewLine .....	725
Description .....	725
Returns .....	725
Syntax .....	725
Parameters .....	726

---

NewMath	726
Description	726
Returns	726
Syntax	726
Parameters	726
NewMathML	726
Description	726
Returns	727
Syntax	727
Parameters	727
NewPolyline	727
Description	727
Returns	727
Syntax	727
Parameters	727
NewPolygon	728
Description	728
Returns	728
Syntax	728
Parameters	728
NewRectangle	728
Description	728
Returns	728
Syntax	728
Parameters	729
NewRoundRect	729
Description	729
Returns	729
Syntax	729
Parameters	729
NewTextFrame	729
Description	729
Returns	730
Syntax	730
Parameters	730
NewTextLine	730
Description	730
Returns	730
Syntax	730
Parameters	730
NewUnanchoredFrame	731
Description	731
Returns	731
Syntax	731
Parameters	731
NewAnchoredFormattedObject	731



---

Description .....	731
Returns .....	732
Syntax .....	732
Parameters .....	732
NewAnchoredFormattedVar .....	732
Description .....	732
Returns .....	733
Syntax .....	733
Parameters .....	733
NewAnchoredFormattedXRef .....	733
Description .....	733
Returns .....	733
Syntax .....	733
Parameters .....	733
NewAnchoredFormattedTbl .....	734
Description .....	734
Returns .....	734
Syntax .....	734
Parameters .....	734
NewAnchoredObject .....	734
Description .....	734
Returns .....	735
Syntax .....	735
Parameters .....	735
NewAnchoredAFrame .....	735
Description .....	735
Returns .....	736
Syntax .....	736
Parameters .....	736
NewAnchoredFn .....	736
Description .....	736
Returns .....	736
Syntax .....	736
Parameters .....	736
NewAnchoredMarker .....	736
Description .....	736
Returns .....	737
Syntax .....	737
Parameters .....	737
NewAnchoredTiApiClient .....	737
Description .....	737
Returns .....	737
Syntax .....	737
Parameters .....	737
NewAnchoredTbl .....	737
Description .....	737

---

Returns	738
Syntax	738
Parameters	738
NewSeriesObject	738
Description	738
Returns	739
Syntax	739
Parameters	739
NewSeriesBodyPage	739
Description	739
Returns	739
Syntax	739
Parameters	740
NewSeriesPgf	740
Description	740
Returns	740
Syntax	740
Parameters	740
GetNamedObject	740
Description	740
Returns	741
Syntax	741
Parameters	742
GetNamedAttrCondExpr	742
Description	742
Returns	742
Syntax	742
Parameters	742
GetNamedCharFmt	742
Description	742
Returns	742
Syntax	743
Parameters	743
GetNamedColor	743
Description	743
Returns	743
Syntax	743
Parameters	743
GetNamedCombinedFontDefn	743
Description	743
Returns	743
Syntax	744
Parameters	744
GetNamedCondFmt	744
Description	744
Returns	744

---

Syntax	744
Parameters	744
GetNamedElementDef	744
Description	744
Returns	744
Syntax	745
Parameters	745
GetNamedFmtChangeList	745
Description	745
Returns	745
Syntax	745
Parameters	745
GetNamedMasterPage	745
Description	745
Returns	745
Syntax	746
Parameters	746
GetNamedPgfmt	746
Description	746
Returns	746
Syntax	746
Parameters	746
GetNamedRefPage	746
Description	746
Returns	746
Syntax	747
Parameters	747
GetNamedRulingFmt	747
Description	747
Returns	747
Syntax	747
Parameters	747
GetNamedTblFmt	747
Description	747
Returns	747
Syntax	748
Parameters	748
GetNamedUnanchoredFrame	748
Description	748
Returns	748
Syntax	748
Parameters	748
GetNamedVarFmt	748
Description	748
Returns	748
Syntax	749

---

Parameters .....	749
GetNamedXRefFmt .....	749
Description .....	749
Returns .....	749
Syntax .....	749
Parameters .....	749
GetNamedMarkerType .....	749
Description .....	749
Returns .....	749
Syntax .....	750
Parameters .....	750
NewNamedObject .....	750
Description .....	750
Returns .....	751
Syntax .....	751
Parameters .....	751
NewNamedAttrCondExpr .....	751
Description .....	751
Returns .....	751
Syntax .....	752
Parameters .....	752
NewNamedCharFmt .....	752
Description .....	752
Returns .....	752
Syntax .....	752
Parameters .....	752
NewNamedCombinedFontDefn .....	752
Description .....	752
Returns .....	752
Syntax .....	753
Parameters .....	753
NewNamedColor .....	753
Description .....	753
Returns .....	753
Syntax .....	753
Parameters .....	753
NewNamedCondFmt .....	753
Description .....	753
Returns .....	753
Syntax .....	754
Parameters .....	754
NewNamedElementDef .....	754
Description .....	754
Returns .....	754
Syntax .....	754
Parameters .....	754

---

NewNamedFmtChangeList . . . . .	754
Description . . . . .	754
Returns . . . . .	754
Syntax . . . . .	755
Parameters . . . . .	755
NewNamedMasterPage . . . . .	755
Description . . . . .	755
Returns . . . . .	755
Syntax . . . . .	755
Parameters . . . . .	755
NewNamedPgfmt . . . . .	755
Description . . . . .	755
Returns . . . . .	755
Syntax . . . . .	756
Parameters . . . . .	756
NewNamedRefPage . . . . .	756
Description . . . . .	756
Returns . . . . .	756
Syntax . . . . .	756
Parameters . . . . .	756
NewNamedRulingFmt . . . . .	756
Description . . . . .	756
Returns . . . . .	756
Syntax . . . . .	757
Parameters . . . . .	757
NewNamedTblFmt . . . . .	757
Description . . . . .	757
Returns . . . . .	757
Syntax . . . . .	757
Parameters . . . . .	757
NewNamedVarFmt . . . . .	757
Description . . . . .	757
Returns . . . . .	757
Syntax . . . . .	758
Parameters . . . . .	758
NewNamedXRefFmt . . . . .	758
Description . . . . .	758
Returns . . . . .	758
Syntax . . . . .	758
Parameters . . . . .	758
NewNamedMarkerType . . . . .	758
Description . . . . .	758
Returns . . . . .	758
Syntax . . . . .	759
Parameters . . . . .	759
DeleteUnusedFmts . . . . .	759

---

Description .....	759
Returns .....	759
Syntax .....	759
Parameters .....	759
DeleteUnusedCharFmt .....	760
Description .....	760
Returns .....	760
Syntax .....	760
DeleteUnusedPgFmt .....	760
Description .....	760
Returns .....	760
Syntax .....	760
DeleteUnusedTblFmt .....	760
Description .....	760
Returns .....	760
Syntax .....	760
Find .....	761
Description .....	761
Returns .....	764
Syntax .....	764
Parameters .....	764
SetElementRange .....	764
Description .....	764
Returns .....	765
Syntax .....	765
Parameters .....	765
ElementLocToTextLoc .....	765
Description .....	765
Returns .....	765
Syntax .....	766
Parameters .....	766
UpdateVariables .....	766
Description .....	766
Returns .....	766
Syntax .....	766
TrackChangesAcceptAll .....	766
Description .....	766
Returns .....	767
Syntax .....	767
TrackChangesRejectAll .....	767
Description .....	767
Returns .....	767
Syntax .....	767
UpdateXRef .....	767
Description .....	767
Returns .....	768

---

Syntax	768
Parameters	768
UpdateDITAReference	769
Description	769
UpdateDITARefereces	770
Description	770
Returns	770
Syntax	770
Parameters	770
Save	771
Description	771
Returns	771
Syntax	771
Parameters	771
Import	771
Description	771
Returns	771
Syntax	771
Parameters	772
AddNewBuildExpr	772
Description	772
Returns	772
Syntax	772
Parameters	773
DeleteBuildExpr	773
Description	773
Returns	773
Syntax	773
Parameters	773
SetActiveBuildExpr	774
Description	774
Returns	774
Syntax	774
Parameters	774
GetActiveBuildExpr	774
Description	774
Returns	775
Syntax	775
GetBuildExpr	775
Description	775
Returns	775
Syntax	775
Parameters	775
GetBuildExprCatalog	776
Description	776
Returns	776

---

Syntax	776
GetProps	776
Description	776
Returns	776
Syntax	776
SetProps	776
Description	776
Returns	776
Syntax	777
Parameters	777
ObjectValid	777
Description	777
Returns	777
Syntax	777
Element	777
GetAttributes	777
Description	777
Returns	778
Syntax	778
Delete	779
Description	779
Returns	779
Syntax	779
UpdateDITAResferences	779
GetText	779
Description	779
Returns	779
Syntax	779
Parameters	779
GetProps	779
Description	779
Returns	780
Syntax	780
SetProps	780
Description	780
Returns	780
Syntax	780
Parameters	780
ObjectValid	780
Description	780
Returns	780
Syntax	780
ElementCatalogEntries	781
concat	781
Description	781
Returns	781



---

Syntax	781
Parameters	781
pop	781
Description	781
Returns	781
Syntax	781
push	782
Description	782
Returns	782
Syntax	782
Parameters	782
ElementCatalogEntry	782
ElementCatalogEntry	782
Description	782
Returns	782
Syntax	782
Parameters	783
ElementDef	783
NewTextFmtRules	783
Description	783
Returns	784
Syntax	784
NewObjectFmtRules	784
Description	784
Returns	785
Syntax	785
NewPrefixRules	785
Description	785
Returns	785
Syntax	785
NewSuffixRules	786
Description	786
Returns	786
Syntax	786
NewFirstPgfrRules	786
Description	786
Returns	787
Syntax	787
NewLastPgfrRules	787
Description	787
Returns	788
Syntax	788
ElementDefIsText	788
Description	788
Returns	788
Syntax	788

---

NewElement . . . . .	788
Description . . . . .	788
Returns . . . . .	789
Syntax . . . . .	790
Parameters . . . . .	790
NewElementInHierarchy . . . . .	790
Description . . . . .	790
Returns . . . . .	791
Syntax . . . . .	791
Parameters . . . . .	791
WrapElement . . . . .	791
Description . . . . .	791
Returns . . . . .	792
Syntax . . . . .	792
GetAttributeDefs . . . . .	792
Description . . . . .	792
Returns . . . . .	793
Syntax . . . . .	793
Delete . . . . .	793
Description . . . . .	793
Returns . . . . .	794
Syntax . . . . .	794
GetProps . . . . .	794
Description . . . . .	794
Returns . . . . .	794
Syntax . . . . .	794
SetProps . . . . .	794
Description . . . . .	794
Returns . . . . .	794
Syntax . . . . .	794
Parameters . . . . .	794
ObjectValid . . . . .	795
Description . . . . .	795
Returns . . . . .	795
Syntax . . . . .	795
ElementLoc . . . . .	795
ElementLoc . . . . .	795
Description . . . . .	795
Returns . . . . .	795
Syntax . . . . .	795
Parameters . . . . .	795
ElementRange . . . . .	796
ElementRange . . . . .	796
Description . . . . .	796
Returns . . . . .	796
Syntax . . . . .	796

---

Parameters	796
Ellipse	796
Delete	796
Description	796
Returns	796
Syntax	796
GetProps	797
Description	797
Returns	797
Syntax	797
SetProps	797
Description	797
Returns	797
Syntax	797
Parameters	797
ObjectValid	797
Description	797
Returns	798
Syntax	798
FilterArgs	798
FilterArgs	798
Description	798
Returns	798
Definition	798
Parameters	798
Flow	799
Delete	799
Description	799
Returns	799
Syntax	799
GetText	799
Description	799
Returns	799
Syntax	799
Parameters	799
GetProps	799
Description	799
Returns	800
Syntax	800
SetProps	800
Description	800
Returns	800
Syntax	800
Parameters	800
ObjectValid	800
Description	800

---

Returns	800
Syntax	800
FmtChangeList	801
Delete	801
Description	801
Returns	801
Syntax	801
GetProps	801
Description	801
Returns	801
Syntax	801
SetProps	801
Description	801
Returns	801
Syntax	801
Parameters	802
ObjectValid	802
Description	802
Returns	802
Syntax	802
FmtRule	802
NewFmtRuleClauses	802
Description	802
Returns	803
Syntax	803
Delete	803
Description	803
Returns	803
Syntax	803
GetProps	803
Description	803
Returns	803
Syntax	803
SetProps	804
Description	804
Returns	804
Syntax	804
Parameters	804
ObjectValid	804
Description	804
Returns	804
Syntax	804
FmtRuleClause	804
NewSubFmtRule	804
Description	804
Returns	805

---

Syntax	805
NewFmtChangeList	805
Description	805
Returns	806
Syntax	806
Delete	806
Description	806
Returns	806
Syntax	806
GetProps	806
Description	806
Returns	806
Syntax	807
SetProps	807
Description	807
Returns	807
Syntax	807
Parameters	807
ObjectValid	807
Description	807
Returns	807
Syntax	807
Fn	808
Delete	808
Description	808
Returns	808
Syntax	808
GetText	808
Description	808
Returns	808
Syntax	808
Parameters	808
GetProps	808
Description	808
Returns	809
Syntax	809
SetProps	809
Description	809
Returns	809
Syntax	809
Parameters	809
ObjectValid	809
Description	809
Returns	809
Syntax	809
Font	810

---

Font . . . . .	810
Description . . . . .	810
Returns . . . . .	810
Syntax . . . . .	810
Parameters . . . . .	810
Fonts . . . . .	810
concat . . . . .	810
Description . . . . .	810
Returns . . . . .	811
Syntax . . . . .	811
Parameters . . . . .	811
pop . . . . .	811
Description . . . . .	811
Returns . . . . .	811
Syntax . . . . .	811
push . . . . .	811
Description . . . . .	811
Returns . . . . .	811
Syntax . . . . .	811
Parameters . . . . .	812
Group . . . . .	812
Delete . . . . .	812
Description . . . . .	812
Returns . . . . .	812
Syntax . . . . .	812
GetProps . . . . .	812
Description . . . . .	812
Returns . . . . .	812
Syntax . . . . .	812
SetProps . . . . .	813
Description . . . . .	813
Returns . . . . .	813
Syntax . . . . .	813
Parameters . . . . .	813
ObjectValid . . . . .	813
Description . . . . .	813
Returns . . . . .	813
Syntax . . . . .	813
HiddenPage . . . . .	813
GetProps . . . . .	813
Description . . . . .	813
Returns . . . . .	814
Syntax . . . . .	814
SetProps . . . . .	814
Description . . . . .	814
Returns . . . . .	814

---

Syntax	814
Parameters	814
ObjectValid	814
Description	814
Returns	814
Syntax	814
ApplyPageLayout	815
Description	815
Returns	815
Syntax	815
Parameters	815
Inset	816
GetIntByName	816
Description	816
Returns	816
Syntax	816
Parameters	816
GetMetricByName	817
Description	817
Returns	817
Syntax	817
Parameters	817
GetUBytesByName	818
Description	818
Returns	818
Syntax	818
Parameters	819
SetIntByName	819
Description	819
Returns	819
Syntax	819
Parameters	820
SetMetricByName	820
Description	820
Returns	820
Syntax	821
Parameters	821
SetUBytesByName	821
Description	821
Returns	821
Syntax	821
Parameters	822
Delete	822
Description	822
Returns	822
Syntax	822

---

DeletePropByName . . . . .	822
Description . . . . .	822
Returns . . . . .	823
Syntax . . . . .	823
Parameters . . . . .	823
GetProps . . . . .	823
Description . . . . .	823
Returns . . . . .	823
Syntax . . . . .	823
SetProps . . . . .	823
Description . . . . .	823
Returns . . . . .	823
Syntax . . . . .	823
Parameters . . . . .	824
ObjectValid . . . . .	824
Description . . . . .	824
Returns . . . . .	824
Syntax . . . . .	824
Ints . . . . .	824
concat . . . . .	824
Description . . . . .	824
Returns . . . . .	824
Syntax . . . . .	824
Parameters . . . . .	825
pop . . . . .	825
Description . . . . .	825
Returns . . . . .	825
Syntax . . . . .	825
push . . . . .	825
Description . . . . .	825
Returns . . . . .	825
Syntax . . . . .	825
Parameters . . . . .	825
Iterator . . . . .	826
GetNextIteratorItem . . . . .	826
Description . . . . .	826
Returns . . . . .	826
Syntax . . . . .	826
Delete . . . . .	826
Description . . . . .	826
Returns . . . . .	826
Syntax . . . . .	826
GetProps . . . . .	826
Description . . . . .	826
Returns . . . . .	826
Syntax . . . . .	826



---

SetProps . . . . .	827
Description . . . . .	827
Returns . . . . .	827
Syntax . . . . .	827
Parameters . . . . .	827
ObjectValid . . . . .	827
Description . . . . .	827
Returns . . . . .	827
Syntax . . . . .	827
KeyCatalog . . . . .	827
NewKeyDefinition . . . . .	827
Description . . . . .	827
Returns . . . . .	828
Syntax . . . . .	828
Parameters . . . . .	828
UpdateKeyDefinition . . . . .	829
Description . . . . .	829
Returns . . . . .	829
Syntax . . . . .	829
Parameters . . . . .	829
GetKeyDefinition . . . . .	831
Description . . . . .	831
Returns . . . . .	831
Syntax . . . . .	831
Parameters . . . . .	831
GetAllKeyDefinitions . . . . .	832
Description . . . . .	832
Returns . . . . .	832
Syntax . . . . .	832
Parameters . . . . .	833
DeleteAllKeyDefinitions . . . . .	834
Description . . . . .	834
Returns . . . . .	834
Syntax . . . . .	834
GetAllKeys . . . . .	834
Description . . . . .	834
Returns . . . . .	834
Syntax . . . . .	834
Delete . . . . .	835
Description . . . . .	835
Returns . . . . .	835
Syntax . . . . .	835
GetProps . . . . .	835
Description . . . . .	835
Returns . . . . .	835
Syntax . . . . .	835

---

SetProps . . . . .	835
Description . . . . .	835
Returns . . . . .	836
Syntax . . . . .	836
Parameters . . . . .	836
ObjectValid . . . . .	836
Description . . . . .	836
Returns . . . . .	836
Syntax . . . . .	836
Line . . . . .	836
Delete . . . . .	836
Description . . . . .	836
Returns . . . . .	836
Syntax . . . . .	837
GetProps . . . . .	837
Description . . . . .	837
Returns . . . . .	837
Syntax . . . . .	837
SetProps . . . . .	837
Description . . . . .	837
Returns . . . . .	837
Syntax . . . . .	837
Parameters . . . . .	837
ObjectValid . . . . .	837
Description . . . . .	837
Returns . . . . .	838
Syntax . . . . .	838
Marker . . . . .	838
Delete . . . . .	838
Description . . . . .	838
Returns . . . . .	838
Syntax . . . . .	838
GetProps . . . . .	838
Description . . . . .	838
Returns . . . . .	838
Syntax . . . . .	838
SetProps . . . . .	838
Description . . . . .	838
Returns . . . . .	839
Syntax . . . . .	839
Parameters . . . . .	839
ObjectValid . . . . .	839
Description . . . . .	839
Returns . . . . .	839
Syntax . . . . .	839
MarkerType . . . . .	839

---

Delete . . . . .	839
Description . . . . .	839
Returns . . . . .	839
Syntax . . . . .	840
GetProps . . . . .	840
Description . . . . .	840
Returns . . . . .	840
Syntax . . . . .	840
SetProps . . . . .	840
Description . . . . .	840
Returns . . . . .	840
Syntax . . . . .	840
Parameters . . . . .	840
ObjectValid . . . . .	840
Description . . . . .	840
Returns . . . . .	841
Syntax . . . . .	841
MasterPage . . . . .	841
Delete . . . . .	841
Description . . . . .	841
Returns . . . . .	841
Syntax . . . . .	841
GetProps . . . . .	841
Description . . . . .	841
Returns . . . . .	841
Syntax . . . . .	841
SetProps . . . . .	841
Description . . . . .	841
Returns . . . . .	842
Syntax . . . . .	842
Parameters . . . . .	842
ObjectValid . . . . .	842
Description . . . . .	842
Returns . . . . .	842
Syntax . . . . .	842
ApplyPageLayout . . . . .	842
Description . . . . .	842
Returns . . . . .	842
Syntax . . . . .	843
Parameters . . . . .	843
Math . . . . .	843
Delete . . . . .	843
Description . . . . .	843
Returns . . . . .	843
Syntax . . . . .	843
GetProps . . . . .	843

---

Description .....	843
Returns .....	843
Syntax .....	843
SetProps .....	844
Description .....	844
Returns .....	844
Syntax .....	844
Parameters .....	844
ObjectValid .....	844
Description .....	844
Returns .....	844
Syntax .....	844
MathML .....	844
Delete .....	844
Description .....	844
Returns .....	845
Syntax .....	845
GetProps .....	845
Description .....	845
Returns .....	845
Syntax .....	845
SetProps .....	845
Description .....	845
Returns .....	845
Syntax .....	845
Parameters .....	845
ObjectValid .....	846
Description .....	846
Returns .....	846
Syntax .....	846
Menu .....	846
DefineAndAddCommand .....	846
Description .....	846
Returns .....	847
Syntax .....	847
Parameters .....	848
AddCommandToMenu .....	848
Description .....	848
Returns .....	849
Syntax .....	849
Parameters .....	849
AddMenuToMenu .....	849
Description .....	849
Returns .....	850
Syntax .....	850
Parameters .....	851

---

DefineAndAddMenu . . . . .	851
Description . . . . .	851
Returns . . . . .	852
Syntax . . . . .	852
Parameters . . . . .	853
MenuItemInMenu . . . . .	853
Description . . . . .	853
Returns . . . . .	853
Syntax . . . . .	853
Parameters . . . . .	854
Delete . . . . .	854
Description . . . . .	854
Returns . . . . .	854
Syntax . . . . .	854
GetProps . . . . .	854
Description . . . . .	854
Returns . . . . .	854
Syntax . . . . .	854
SetProps . . . . .	855
Description . . . . .	855
Returns . . . . .	855
Syntax . . . . .	855
Parameters . . . . .	855
ObjectValid . . . . .	855
Description . . . . .	855
Returns . . . . .	855
Syntax . . . . .	855
Delete . . . . .	855
Description . . . . .	855
Returns . . . . .	856
Syntax . . . . .	856
MenuItemSeparator . . . . .	856
GetProps . . . . .	856
Description . . . . .	856
Returns . . . . .	856
Syntax . . . . .	856
SetProps . . . . .	856
Description . . . . .	856
Returns . . . . .	856
Syntax . . . . .	856
Parameters . . . . .	857
ObjectValid . . . . .	857
Description . . . . .	857
Returns . . . . .	857
Syntax . . . . .	857
Delete . . . . .	857

---

Description .....	857
Returns .....	857
Syntax .....	857
Metrics .....	857
concat .....	858
Description .....	858
Returns .....	858
Syntax .....	858
Parameters .....	858
pop .....	858
Description .....	858
Returns .....	858
Syntax .....	858
push .....	858
Description .....	858
Returns .....	858
Syntax .....	859
Parameters .....	859
Pgf .....	859
Delete .....	859
Description .....	859
Returns .....	859
Syntax .....	859
GetText .....	859
Description .....	859
Returns .....	859
Syntax .....	859
Parameters .....	860
GetProps .....	860
Description .....	860
Returns .....	860
Syntax .....	860
SetProps .....	860
Description .....	860
Returns .....	860
Syntax .....	860
Parameters .....	860
ObjectValid .....	861
Description .....	861
Returns .....	861
Syntax .....	861
PgfFmt .....	861
Delete .....	861
Description .....	861
Returns .....	861
Syntax .....	861

---

GetProps . . . . .	861
Description . . . . .	861
Returns . . . . .	861
Syntax . . . . .	861
SetProps . . . . .	862
Description . . . . .	862
Returns . . . . .	862
Syntax . . . . .	862
Parameters . . . . .	862
ObjectValid . . . . .	862
Description . . . . .	862
Returns . . . . .	862
Syntax . . . . .	862
Point . . . . .	863
Point . . . . .	863
Description . . . . .	863
Returns . . . . .	863
Syntax . . . . .	863
Parameters . . . . .	863
Points . . . . .	863
concat . . . . .	863
Description . . . . .	863
Returns . . . . .	863
Syntax . . . . .	863
Parameters . . . . .	864
pop . . . . .	864
Description . . . . .	864
Returns . . . . .	864
Syntax . . . . .	864
push . . . . .	864
Description . . . . .	864
Returns . . . . .	864
Syntax . . . . .	864
Parameters . . . . .	864
Polygon . . . . .	865
Delete . . . . .	865
Description . . . . .	865
Returns . . . . .	865
Syntax . . . . .	865
GetProps . . . . .	865
Description . . . . .	865
Returns . . . . .	865
Syntax . . . . .	865
SetProps . . . . .	865
Description . . . . .	865
Returns . . . . .	865

---

Syntax	865
Parameters	866
ObjectValid	866
Description	866
Returns	866
Syntax	866
Polyline	866
Delete	866
Description	866
Returns	866
Syntax	866
GetProps	866
Description	866
Returns	867
Syntax	867
SetProps	867
Description	867
Returns	867
Syntax	867
Parameters	867
ObjectValid	867
Description	867
Returns	867
Syntax	867
PropIdent	868
PropIdent	868
Description	868
Returns	868
Syntax	868
Parameters	868
PropVal	868
PropVal	868
Description	868
Returns	868
Syntax	868
Parameters	869
PropVals	869
concat	869
Description	869
Returns	869
Syntax	869
Parameters	869
pop	869
Description	869
Returns	870
Syntax	870



---

push . . . . .	870
Description . . . . .	870
Returns . . . . .	870
Syntax . . . . .	870
Parameters . . . . .	870
Rectangle . . . . .	870
Delete . . . . .	870
Description . . . . .	870
Returns . . . . .	870
Syntax . . . . .	870
GetProps . . . . .	871
Description . . . . .	871
Returns . . . . .	871
Syntax . . . . .	871
SetProps . . . . .	871
Description . . . . .	871
Returns . . . . .	871
Syntax . . . . .	871
Parameters . . . . .	871
ObjectValid . . . . .	871
Description . . . . .	871
Returns . . . . .	872
Syntax . . . . .	872
RefPage . . . . .	872
Delete . . . . .	872
Description . . . . .	872
Returns . . . . .	872
Syntax . . . . .	872
GetProps . . . . .	872
Description . . . . .	872
Returns . . . . .	872
Syntax . . . . .	872
SetProps . . . . .	872
Description . . . . .	872
Returns . . . . .	873
Syntax . . . . .	873
Parameters . . . . .	873
ObjectValid . . . . .	873
Description . . . . .	873
Returns . . . . .	873
Syntax . . . . .	873
ApplyPageLayout . . . . .	873
Description . . . . .	873
Returns . . . . .	873
Syntax . . . . .	873
Parameters . . . . .	874

---

RoundRect . . . . .	874
Delete . . . . .	874
Description . . . . .	874
Returns . . . . .	874
Syntax . . . . .	874
GetProps . . . . .	874
Description . . . . .	874
Returns . . . . .	874
Syntax . . . . .	874
SetProps . . . . .	874
Description . . . . .	874
Returns . . . . .	875
Syntax . . . . .	875
Parameters . . . . .	875
ObjectValid . . . . .	875
Description . . . . .	875
Returns . . . . .	875
Syntax . . . . .	875
Row . . . . .	875
AddRows . . . . .	875
Description . . . . .	875
Returns . . . . .	876
Syntax . . . . .	876
Parameters . . . . .	876
Delete . . . . .	877
Description . . . . .	877
Returns . . . . .	877
Syntax . . . . .	877
GetProps . . . . .	877
Description . . . . .	877
Returns . . . . .	877
Syntax . . . . .	877
SetProps . . . . .	877
Description . . . . .	877
Returns . . . . .	877
Syntax . . . . .	877
Parameters . . . . .	878
ObjectValid . . . . .	878
Description . . . . .	878
Returns . . . . .	878
Syntax . . . . .	878
Rubi . . . . .	878
Delete . . . . .	878
Description . . . . .	878
Returns . . . . .	878
Syntax . . . . .	878

---

GetText2 . . . . .	879
Description . . . . .	879
Returns . . . . .	879
Syntax . . . . .	879
Parameters . . . . .	879
GetProps . . . . .	880
Description . . . . .	880
Returns . . . . .	880
Syntax . . . . .	880
SetProps . . . . .	880
Description . . . . .	880
Returns . . . . .	880
Syntax . . . . .	880
Parameters . . . . .	881
ObjectValid . . . . .	881
Description . . . . .	881
Returns . . . . .	881
Syntax . . . . .	881
RulingFmt . . . . .	881
Delete . . . . .	881
Description . . . . .	881
Returns . . . . .	881
Syntax . . . . .	881
GetProps . . . . .	881
Description . . . . .	881
Returns . . . . .	882
Syntax . . . . .	882
SetProps . . . . .	882
Description . . . . .	882
Returns . . . . .	882
Syntax . . . . .	882
Parameters . . . . .	882
ObjectValid . . . . .	882
Description . . . . .	882
Returns . . . . .	882
Syntax . . . . .	882
Strings . . . . .	883
concat . . . . .	883
Description . . . . .	883
Returns . . . . .	883
Syntax . . . . .	883
Parameters . . . . .	883
pop . . . . .	883
Description . . . . .	883
Returns . . . . .	883
Syntax . . . . .	883

---

push . . . . .	883
Description . . . . .	883
Returns . . . . .	884
Syntax . . . . .	884
Parameters . . . . .	884
SubCol . . . . .	884
Delete . . . . .	884
Description . . . . .	884
Returns . . . . .	884
Syntax . . . . .	884
GetText . . . . .	884
Description . . . . .	884
Returns . . . . .	884
Syntax . . . . .	884
Parameters . . . . .	885
GetProps . . . . .	885
Description . . . . .	885
Returns . . . . .	885
Syntax . . . . .	885
SetProps . . . . .	885
Description . . . . .	885
Returns . . . . .	885
Syntax . . . . .	885
Parameters . . . . .	885
ObjectValid . . . . .	886
Description . . . . .	886
Returns . . . . .	886
Syntax . . . . .	886
Tab . . . . .	886
Tab . . . . .	886
Description . . . . .	886
Returns . . . . .	887
Syntax . . . . .	887
Parameters . . . . .	887
Tabs . . . . .	887
concat . . . . .	887
Description . . . . .	887
Returns . . . . .	887
Syntax . . . . .	887
Parameters . . . . .	888
pop . . . . .	888
Description . . . . .	888
Returns . . . . .	888
Syntax . . . . .	888
push . . . . .	888
Description . . . . .	888

---

Returns	888
Syntax	888
Parameters	888
Tbl	889
AddCols	889
Description	889
Returns	889
Syntax	889
Parameters	889
DeleteCols	890
Description	890
Returns	890
Syntax	890
Parameters	891
DeleteRows	891
Description	891
Returns	891
Syntax	891
Parameters	892
GetTableColumnByColNum	892
Description	892
Returns	892
Syntax	892
Parameters	893
MakeTblSelection	893
Description	893
Returns	893
Syntax	894
Parameters	894
Delete	894
Description	894
Returns	894
Syntax	894
GetProps	894
Description	894
Returns	895
Syntax	895
SetProps	895
Description	895
Returns	895
Syntax	895
Parameters	895
ObjectValid	895
Description	895
Returns	895
Syntax	895

---

TblFmt . . . . .	896
Delete . . . . .	896
GetProps . . . . .	896
SetProps . . . . .	896
ObjectValid . . . . .	896
TextFrame . . . . .	896
Delete . . . . .	896
Description . . . . .	896
Returns . . . . .	896
Syntax . . . . .	896
GetText . . . . .	896
Description . . . . .	896
Returns . . . . .	897
Syntax . . . . .	897
Parameters . . . . .	897
GetProps . . . . .	897
Description . . . . .	897
Returns . . . . .	897
Syntax . . . . .	897
SetProps . . . . .	897
Description . . . . .	897
Returns . . . . .	897
Syntax . . . . .	897
Parameters . . . . .	898
ObjectValid . . . . .	898
Description . . . . .	898
Returns . . . . .	898
Syntax . . . . .	898
TextItems . . . . .	898
concat . . . . .	898
Description . . . . .	898
Returns . . . . .	898
Syntax . . . . .	898
Parameters . . . . .	899
pop . . . . .	899
Description . . . . .	899
Returns . . . . .	899
Syntax . . . . .	899
push . . . . .	899
Description . . . . .	899
Returns . . . . .	899
Syntax . . . . .	899
Parameters . . . . .	899
TextLine . . . . .	900
Delete . . . . .	900
Description . . . . .	900

---

Returns	900
Syntax	900
GetText	900
Description	900
Returns	900
Syntax	900
Parameters	900
GetProps	900
Description	900
Returns	901
Syntax	901
SetProps	901
Description	901
Returns	901
Syntax	901
Parameters	901
ObjectValid	901
Description	901
Returns	901
Syntax	901
TextLoc	902
TextLoc	902
Description	902
Returns	902
Syntax	902
Parameters	902
TextRange	902
TextRange	902
Description	902
Returns	902
Syntax	902
Parameters	903
TiApiClient	903
GetText	903
GetProps	903
SetProps	903
ObjectValid	903
DeleteTextInsetContents	903
Description	903
Returns	904
Syntax	904
UpdateTextInset	904
Description	904
Returns	905
Syntax	905
ConvertToText	905

---

Description .....	905
Returns .....	905
Syntax .....	905
Delete .....	905
DeletePropByName .....	905
Description .....	905
Returns .....	906
Syntax .....	906
Parameters .....	906
TiFlow .....	906
GetText .....	906
GetProps .....	906
SetProps .....	906
ObjectValid .....	906
DeleteTextInsetContents .....	906
UpdateTextInset .....	907
ConvertToText .....	907
Description .....	907
Returns .....	907
Syntax .....	907
Delete .....	907
DeletePropByName .....	907
TiText .....	907
GetText .....	907
GetProps .....	907
SetProps .....	907
ObjectValid .....	908
DeleteTextInsetContents .....	908
UpdateTextInset .....	908
ConvertToText .....	908
Description .....	908
Returns .....	908
Syntax .....	908
Delete .....	908
DeletePropByName .....	908
TiTextTable .....	908
GetText .....	908
GetProps .....	909
SetProps .....	909
ObjectValid .....	909
DeleteTextInsetContents .....	909
UpdateTextInset .....	909
ConvertToText .....	909
Description .....	909
Returns .....	909
Syntax .....	909



---

Delete	909
DeletePropByName	909
TypedVals	910
concat	910
Description	910
Returns	910
Syntax	910
Parameters	910
pop	910
Description	910
Returns	910
Syntax	910
push	910
Description	910
Returns	911
Syntax	911
Parameters	911
UBytes	911
concat	911
Description	911
Returns	911
Syntax	911
Parameters	911
pop	912
Description	912
Returns	912
Syntax	912
push	912
Description	912
Returns	912
Syntax	912
Parameters	912
UInts	912
concat	912
Description	912
Returns	913
Syntax	913
Parameters	913
pop	913
Description	913
Returns	913
Syntax	913
push	913
Description	913
Returns	913
Syntax	913

---

Parameters	914
UnanchoredFrame	914
Delete	914
Description	914
Returns	914
Syntax	914
GetProps	914
Description	914
Returns	914
Syntax	914
SetProps	914
Description	914
Returns	915
Syntax	915
Parameters	915
ObjectValid	915
Description	915
Returns	915
Syntax	915
Var	915
Delete	915
Description	915
Returns	915
Syntax	916
GetText	916
Description	916
Returns	916
Syntax	916
Parameters	916
GetProps	916
Description	916
Returns	916
Syntax	916
SetProps	916
Description	916
Returns	917
Syntax	917
Parameters	917
ObjectValid	917
Description	917
Returns	917
Syntax	917
VarFmt	917
Delete	917
Description	917
Returns	917

---

Syntax	918
GetProps	918
Description	918
Returns	918
Syntax	918
SetProps	918
Description	918
Returns	918
Syntax	918
Parameters	918
ObjectValid	918
Description	918
Returns	919
Syntax	919
XRef	919
Delete	919
Description	919
Returns	919
Syntax	919
GetText	919
Description	919
Returns	919
Syntax	919
Parameters	920
GetProps	920
Description	920
Returns	920
Syntax	920
SetProps	920
Description	920
Returns	920
Syntax	920
Parameters	920
ObjectValid	921
Description	921
Returns	921
Syntax	921
XRefFmt	921
Delete	921
Description	921
Returns	921
Syntax	921
GetProps	921
Description	921
Returns	921
Syntax	921

---

SetProps . . . . .	922
Description . . . . .	922
Returns . . . . .	922
Syntax . . . . .	922
Parameters . . . . .	922
ObjectValid . . . . .	922
Description . . . . .	922
Returns . . . . .	922
Syntax . . . . .	922
app . . . . .	922
Close . . . . .	922
GetNamedObject . . . . .	923
Description . . . . .	923
Returns . . . . .	923
Syntax . . . . .	923
Parameters . . . . .	923
GetNamedBook . . . . .	923
Description . . . . .	923
Returns . . . . .	924
Syntax . . . . .	924
Parameters . . . . .	924
GetNamedCommand . . . . .	924
Description . . . . .	924
Returns . . . . .	924
Syntax . . . . .	924
Parameters . . . . .	925
GetNamedMenu . . . . .	925
Description . . . . .	925
Returns . . . . .	925
Syntax . . . . .	925
Parameters . . . . .	925
GetNamedMenuItemSeparator . . . . .	925
Description . . . . .	925
Returns . . . . .	926
Syntax . . . . .	926
Parameters . . . . .	926
NewNamedObject . . . . .	926
Description . . . . .	926
Returns . . . . .	926
Syntax . . . . .	926
Parameters . . . . .	927
NewNamedBook . . . . .	927
Description . . . . .	927
Returns . . . . .	927
Syntax . . . . .	927
Parameters . . . . .	927

---

NewNamedCommand . . . . .	927
Description . . . . .	927
Returns . . . . .	928
Syntax . . . . .	928
Parameters . . . . .	928
NewNamedMenu . . . . .	928
Description . . . . .	928
Returns . . . . .	928
Syntax . . . . .	928
Parameters . . . . .	929
NewNamedMenuItemSeparator . . . . .	929
Description . . . . .	929
Returns . . . . .	929
Syntax . . . . .	929
Parameters . . . . .	929
GetProps . . . . .	929
SetProps . . . . .	930
ObjectValid . . . . .	930
global . . . . .	930
Alert . . . . .	930
Description . . . . .	930
Returns . . . . .	930
Syntax . . . . .	931
Parameters . . . . .	931
Alive . . . . .	931
Description . . . . .	931
Returns . . . . .	931
Syntax . . . . .	931
CallClient . . . . .	931
Description . . . . .	931
Returns . . . . .	932
Syntax . . . . .	932
Parameters . . . . .	932
Example: XLIFF Export . . . . .	932
Example: XLIFF Import . . . . .	934
CallClientEx . . . . .	935
Description . . . . .	935
Returns . . . . .	936
Syntax . . . . .	936
Parameters . . . . .	936
CheckStatus . . . . .	936
Description . . . . .	936
Returns . . . . .	936
Syntax . . . . .	936
Parameters . . . . .	937
ClientDir . . . . .	937

---

Description .....	937
Returns .....	937
Syntax .....	937
ClientName .....	937
Description .....	937
Returns .....	938
Syntax .....	938
CMSRegister .....	938
Description .....	938
Returns .....	938
Syntax .....	938
Parameters .....	938
CMSGetCmsIdFromName .....	938
Description .....	938
Returns .....	938
Syntax .....	939
Parameters .....	939
CMSAddMenuEntry .....	939
Description .....	939
Returns .....	939
Syntax .....	939
Parameters .....	939
CustomDoc .....	940
Description .....	940
Returns .....	940
Syntax .....	940
Parameters .....	941
DefineMenu .....	941
Description .....	941
Returns .....	942
Syntax .....	942
Parameters .....	942
DisconnectFromSession .....	942
Description .....	942
Returns .....	943
Syntax .....	943
Err .....	943
Description .....	943
Returns .....	943
Syntax .....	943
Parameters .....	943
FamilyFonts .....	943
Description .....	943
Returns .....	943
Syntax .....	943
Parameters .....	944

---

GetKeyCatalog . . . . .	944
Description . . . . .	944
Returns . . . . .	944
Syntax . . . . .	944
Parameters . . . . .	944
GetEncodingForFamily . . . . .	944
Description . . . . .	944
Returns . . . . .	945
Syntax . . . . .	945
Parameters . . . . .	945
GetEncodingForFont . . . . .	945
Description . . . . .	945
Returns . . . . .	946
Syntax . . . . .	946
Parameters . . . . .	946
GetImportDefaultParams . . . . .	946
Description . . . . .	946
Returns . . . . .	946
Syntax . . . . .	946
GetOpenDefaultParams . . . . .	946
Description . . . . .	946
Returns . . . . .	947
Syntax . . . . .	947
GetPropIndex . . . . .	947
Description . . . . .	947
Returns . . . . .	947
Syntax . . . . .	947
Parameters . . . . .	947
GetSaveDefaultParams . . . . .	947
Description . . . . .	947
Returns . . . . .	953
Syntax . . . . .	953
GetSupportedEncodings . . . . .	953
Description . . . . .	953
Returns . . . . .	953
Syntax . . . . .	953
GetUpdateBookDefaultParams . . . . .	954
Description . . . . .	954
Returns . . . . .	954
Syntax . . . . .	954
GetNewXMLDefaultParams . . . . .	954
Description . . . . .	954
Returns . . . . .	954
Syntax . . . . .	954
IsEncodingSupported . . . . .	955
Description . . . . .	955

---

Returns	955
Syntax	955
Parameters	955
LoadMenuCustomizationFile	955
Description	955
Returns	956
Syntax	956
Parameters	956
NewKeyCatalog	957
Description	957
Returns	957
Syntax	957
Parameters	957
NewXML	957
Description	957
Returns	957
Syntax	957
Parameters	958
Notification	958
Description	958
Returns	965
Syntax	965
Parameters	965
PopClipboard	965
Description	965
Returns	965
Syntax	966
Parameters	966
PrintFAErrno	966
Description	966
Returns	966
Syntax	966
Parameters	967
PrintImportStatus	967
Description	967
Returns	967
Syntax	967
Parameters	967
PrintOpenStatus	967
Description	967
Returns	967
Syntax	967
Parameters	968
PrintPropVal	968
Description	968
Returns	968



---

Syntax	968
Parameters	968
PrintPropVals	968
Description	968
Returns	968
Syntax	968
Parameters	969
PrintSaveStatus	969
Description	969
Returns	969
Syntax	969
Parameters	969
PrintTextItem	969
Description	969
Returns	969
Syntax	969
Parameters	970
PrintTextItems	970
Description	970
Returns	970
Syntax	970
Parameters	970
PrintUpdateBookStatus	970
Description	970
Returns	970
Syntax	970
Parameters	971
PushClipboard	971
Description	971
Returns	971
Syntax	971
Parameters	971
ReturnValue	971
Description	971
Returns	973
Syntax	973
Parameters	973
ShutDown	974
Description	974
Returns	974
Syntax	974
Parameters	974
SimpleNewDoc	974
Description	974
Returns	974
Syntax	974

---

Parameters .....	975
UserCancel .....	975
Description .....	975
Returns .....	975
Syntax .....	975
Parameters .....	975
SimpleOpen .....	975
Description .....	975
Returns .....	976
Syntax .....	976
Parameters .....	976
UpdateMenus .....	976
Description .....	976
Returns .....	976
Syntax .....	977
Parameters .....	977
AddMenu .....	977
Description .....	977
Returns .....	977
Syntax .....	977
Parameters .....	978
RemoveMenu .....	978
Description .....	978
Returns .....	978
Syntax .....	978
Parameters .....	978
MenuExists .....	979
Description .....	979
Returns .....	979
Syntax .....	979
Parameters .....	979
ShortcutExists .....	979
Description .....	979
Returns .....	979
Syntax .....	979
Parameters .....	979
GetExportDefaultParams .....	980
Description .....	980
Returns .....	980
Syntax .....	980
Parameters .....	980
PrintExportStatus .....	980
Description .....	980
Returns .....	980
Syntax .....	980
Parameters .....	980

---

ErrorName . . . . .	980
Description . . . . .	980
Returns . . . . .	981
Syntax . . . . .	981
Parameters . . . . .	981
PrintErrno . . . . .	981
Description . . . . .	981
Returns . . . . .	981
Syntax . . . . .	981
Parameters . . . . .	981
NotificationName . . . . .	981
Description . . . . .	981
Returns . . . . .	981
Syntax . . . . .	981
Parameters . . . . .	982
AllocatePropVals . . . . .	982
Description . . . . .	982
Returns . . . . .	982
Syntax . . . . .	982
Parameters . . . . .	982
Open . . . . .	982
Description . . . . .	982
Returns . . . . .	989
Syntax . . . . .	989
Parameters . . . . .	989
ScrollBar . . . . .	990
Description . . . . .	990
Returns . . . . .	990
Syntax . . . . .	990
Parameters . . . . .	990
ChooseFile . . . . .	990
Description . . . . .	990
Returns . . . . .	991
Syntax . . . . .	991
Parameters . . . . .	991
PromptInt . . . . .	992
Description . . . . .	992
Returns . . . . .	992
Syntax . . . . .	992
Parameters . . . . .	992
PromptMetric . . . . .	993
Description . . . . .	993
Returns . . . . .	993
Syntax . . . . .	993
Parameters . . . . .	993
PromptString . . . . .	994

Description .....	994
Returns .....	994
Syntax .....	994
Parameters .....	995
Service . . . . .	995
Description .....	995
Returns .....	995
Syntax .....	995
FileScrollBox . . . . .	995
Description .....	995
Returns .....	995
Syntax .....	995
Parameters .....	996
SetCurrentWorkspace . . . . .	996
Description .....	996
Returns .....	996
Syntax .....	996
Parameters .....	996
GetWorkspaceName . . . . .	996
Description .....	996
Returns .....	996
Syntax .....	996
<b>CMS Connector Framework . . . . .</b>	<b>997</b>
CMS API Data Structures and Enum Constants . . . . .	997
CMSResult . . . . .	997
CMSResult .....	997
CMSItemFileType . . . . .	998
Description .....	998
CMSProperties . . . . .	999
Description .....	999
CMSMenuItem . . . . .	999
Description .....	999
CMSCheckinParam . . . . .	999
Description .....	999
CMSDeleteParam . . . . .	1000
Description .....	1000
CMSInfo . . . . .	1000
Description .....	1000
CMSInfos . . . . .	1000
Description .....	1000
Error Codes . . . . .	1000
CMS API functions . . . . .	1001
APIs to automate CMS connector functionality . . . . .	1002
CMSLogin . . . . .	1002
Description .....	1002

---

Returns	1002
Syntax	1002
Parameters	1003
CMSLogout	1003
Description	1003
Returns	1003
Syntax	1003
Parameters	1003
CMSCheckout	1004
Description	1004
Returns	1004
Syntax	1004
Parameters	1004
CMSCheckin	1004
Description	1004
Returns	1004
Syntax	1004
Parameters	1004
CMSCancelCheckout	1005
Description	1005
Returns	1005
Syntax	1005
Parameters	1005
CMSDelete	1006
Description	1006
Returns	1006
Syntax	1006
Parameters	1006
CMSOpenFile	1006
Description	1006
Returns	1006
Syntax	1006
Parameters	1007
CMSUploadObject	1007
Description	1007
Returns	1007
Syntax	1007
Parameters	1007
CMSDownloadObject	1007
Description	1007
Returns	1008
Syntax	1008
Parameters	1008
GetCMSObjectFromPath	1008
Description	1008
Returns	1008

---

Syntax .....	1008
Parameters .....	1008
<b>Legal notices .....</b>	<b>1009</b>

# Overview

A script is a series of commands that instructs Adobe® FrameMaker® to perform a set of specified actions. Scripts automate repetitive tasks, such as counting the images in a book or adding an indent to all the para tags in a file. Use scripts as a creative tool to streamline time-consuming tasks.

If you are new to scripting, you can acquaint yourself with the basic scripting concepts provided in the [Introduction to Scripting](#) manual.

## About this document

### Naming conventions

To help you identify the structures, constants, and functions defined by the ESTK, this document adheres to the following naming conventions:

Type	Naming Convention	Example
API error codes	Prefix: <code>Constants.FE</code> Suffix: Error number in parentheses	<code>Constants.FE_BadName (-33)</code>
API functions	Suffix: <code>()</code>	<code>UpdateTextInset ()</code>
API scriptable function property names	Prefix: <code>Constants.FS_</code> Suffix: Enumerated value in parentheses	<code>Constants.FS_ForceImportAsText (17)</code>
Flags used by API functions	Prefix: <code>Constants.FF_</code> Suffix: Enumerated value in parentheses	<code>Constants.FF_UFF_VAR ()</code>
Initialization constants	Prefix: <code>Constants.FA_Init</code> Suffix: Enumerated value in parentheses	<code>Constants.FA_Init_First (1)</code>
Notification constants	Prefix: <code>Constants.FA_Note</code>	<code>Constants.FA_Note_PreFileType (9)</code>
Object types	Prefix: <code>Constants.FO_</code> Suffix: Enumerated value in parentheses	<code>Constants.FO_Doc (4)</code>
Property value constants	Prefix: <code>Constants.FV_</code> Suffix: Enumerated value in parentheses	<code>Constants.FV_Doc_Type_MIF (0x03)</code>

## Typographic conventions

Notice the use of different fonts to represent different types of information:

- What you type is shown in `text like this`.
- Function names, property names, structure names, returned values, constants, filter names, program names, paths, and filenames are also shown in `text like this`.

## Scripting support in FrameMaker

FrameMaker supports ExtendScript, which is an extended implementation of JavaScript. While JavaScript files have `.js` extension, ExtendScript files have `.jsx` extension.

For more information on JavaScript modules, tools, utilities, and features that are available to all JavaScript-enabled Adobe applications, see [JavaScript Tools Guide](#).

## Development and debugging tools in ExtendScript toolkit

For assistance in developing, debugging, and testing scripts, Adobe provides the ExtendScript Toolkit. ExtendScript Toolkit is an interactive development and testing environment for ExtendScript, which is installed with FrameMaker and all JavaScript-enabled applications. For details, see Chapter 2, “The ExtendScript Toolkit” and Chapter 8, “ExtendScript Tools and Features” in [JavaScript Tools Guide](#).

## ExtendScript capabilities

Using ExtendScript Toolkit, you can develop and debug ExtendScript. ExtendScript provides the following features that make scripting easy:

- **User interface development tools:** The ScriptUI module defines Window objects that represent platform-specific windows and various control elements, such as buttons and static text. Resource specification allows you to create user-interface elements. Using the event-handling callback functions, you can define the behavior of your user-interaction controls. ScriptUI provides a rich set of containers (such as Group, Panel, and TabbedPanel) and user interface controls (such as Button, Image, Scrollbar, Treeview, and FlashPlayer). ScriptUI works with the ExtendScript JavaScript interpreter to provide JavaScript programs with the ability to create and interact with user interface elements. The Graphic customization objects provide the ability to customize the appearance of user-interface controls before they are drawn. For details, see Chapter 4, “User-Interface Tools” in [JavaScript Tools Guide](#).
- **Extensions:**
  - ExtendScript offers tools for communicating with other computers or the Internet using standard protocols. The Socket object supports low-level TCP connections. For details, see Chapter 6, “External Communication Tools” in [JavaScript Tools Guide](#).
  - Adobe ExtendScript defines classes that simplify cross-platform file-system access. These classes are available to all applications that support a JavaScript interface. For details, see Chapter 3, “File System Access” in [JavaScript Tools Guide](#).



- You can extend the JavaScript DOM for an application. To extend the JavaScript DOM, write a C or C++ shared library, compile it for the relevant platform, and load it into JavaScript as an ExternalObject instance. A shared library is implemented by a DLL in Windows, a bundle or framework in Mac OS, or a SharedObject in UNIX. For details, see Chapter 7, “Integrating External Libraries” in [JavaScript Tools Guide](#).
- **Inter-application communication and messaging:** ExtendScript provides a common scripting environment for all Adobe JavaScript-enabled applications. The cross-DOM API allows implementation of functions, such as open files and execute scripts, that are common across message-enabled applications. Using simple syntax you can implement application-specific functionality for message-enabled application. The BridgeTalk class of the messaging API has globally available static properties and functions that provide access to environmental information relevant for communication between applications. The error codes defined by the interapplication messaging protocol are compatible with the ExtendScript error codes. For details, see Chapter 5, “Interapplication Communication with Scripts” in [JavaScript Tools Guide](#)


## Creating Scripts

Adobe provides ExtendScript Toolkit (ESTK) with its applications as a scripting utility. ESTK has many features that make it easier to use than a text editor. ESTK has a built-in syntax checker that identifies where the problems are in your script and tries to explain how to fix them. You can run your scripts right from the ESTK without saving the file. Running scripts without saving them first saves you time, especially when you are learning to work with ESTK.

## Accessing ESTK

### Connect ESTK to FrameMaker

When you launch ExtendScript Toolkit from FrameMaker (File > Script ...), ExtendScript Toolkit is already connected to FrameMaker. Only when you launch ExtendScript Toolkit from Windows (Start > Programs...), complete the following steps to connect ESTK to FrameMaker.

- 1) In the ExtendScript Toolkit, select Adobe FrameMaker <Version#> in the pop-up menu.
- 2) Click ,
- If FrameMaker is not already running, a message appears: “Target Adobe FrameMaker <Version#> is not running. Do you want to launch Adobe FrameMaker <Version#>?”*
- 3) Click Yes.
- FrameMaker is launched and connected to ESTK.*

## To write, edit, and run scripts

In FrameMaker, do the following:

- 1) Click File > Script > New Script.

*The ESTK editor opens.*

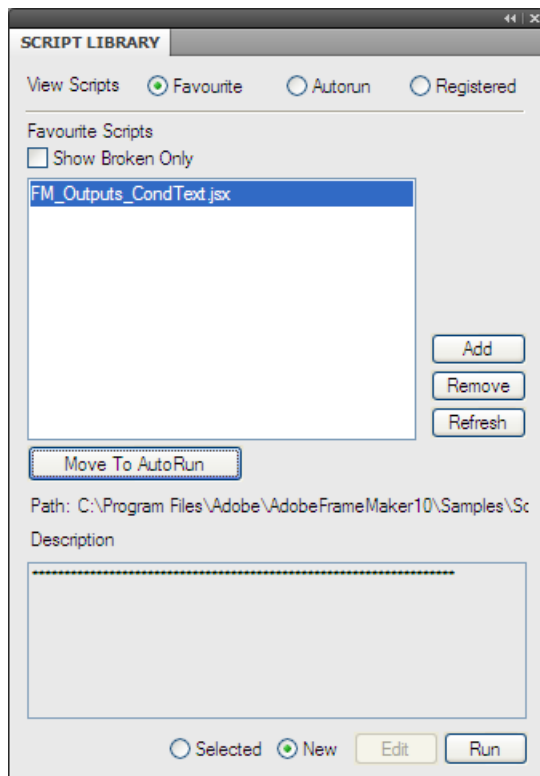
- 2) Write the script in the editor.

- 3) To run the script, click the Play button.

*You can also start ExtendScript Toolkit from Windows (Start > Programs). When you start ExtendScript Toolkit from Windows, connect ESTK to FrameMaker.*

## Using script library

In FrameMaker, select File > Script > Catalog to display the script library.

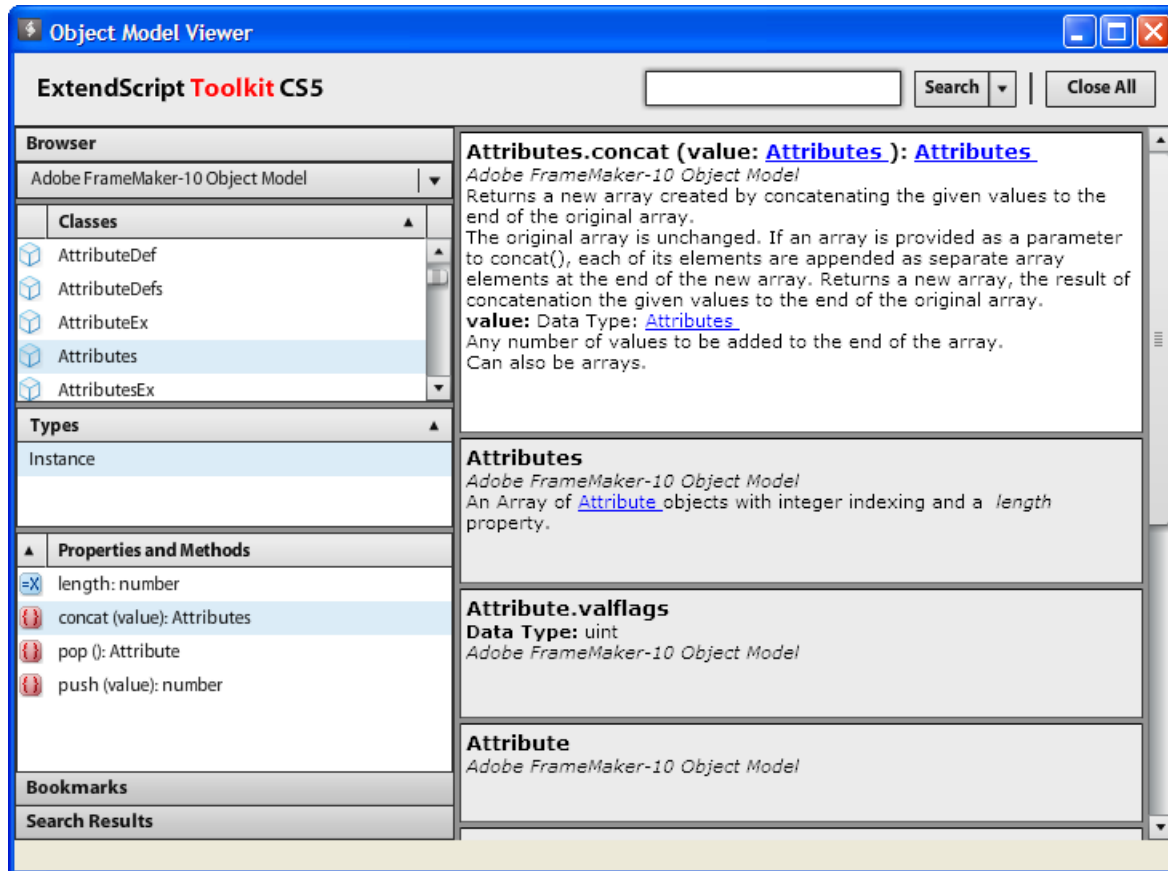


Using the script catalog, you can manage your existing scripts:

- Favorites: Scripts you tag as your favorite appear here.
- Autorun: These scripts run when FrameMaker is launched.
- Registered: Scripts that run on certain triggers, such as on opening a file or saving a file.

## Using object model viewer

Object Model Viewer in ESTK helps you to get the information on different classes and the methods in ExtendScript. Press F1 or click Help > Object Model Viewer to open Object Model Viewer.



In the Object Model Viewer, select an object model under Browse to you can do the following to locate information:

- Under Browser, select the object model for which you want the information. Object Model Viewer displays the classes in the object model and details such as type, properties, method, and description of the class.
- Search for a class, method, or property.
- Bookmark a class, method, or property. It then appears under the Bookmarks section.

## ExtendScript use cases

Some of the ExtendScript use cases can be:

- **Scripts to automate repetitive tasks:** You can write utilities to automate repetitive tasks, such as change alignment of all the tables in a FrameMaker document.
- **Notification-based scripts:** You can write scripts that are triggered on FrameMaker events such as such as opening, printing, or saving a document.

- **Menu command scripts/scripts to extend FrameMaker functionality:** Using these scripts, you can add menus and commands in FrameMaker.
- **Startup scripts:** These scripts get executed when FrameMaker starts. You can set up scripts to run at startup by copying them at the following location: `<FMINSTALL_DIR>\startup`. Alternatively, you can use the Script Library dialog to register Autorun scripts.

# What's new in ESTK

## Export and import XLIFF

The `CallClient` method allows you to export and import XLIFF files. Sample codes for exporting and import XLIFF files are given under the [CallClient](#) method.

## Sample ESTK scripts

ExtendScript is similar to JavaScript. You can easily develop ExtendScript for any of the applications in FrameMaker if you are familiar with JavaScript.

Following are examples for scripts that automate specific activities in FrameMaker.

**NOTE:** Sample scripts shipped with FrameMaker are at the following location: <FMINSTALL\_DIR>\samples

### Adding text to a document and enabling change bars

The following script adds a sample text to a FrameMaker document and then enables the change bar. Open a FrameMaker document before running the script.

Copy and paste the script to ESTK and select FrameMaker <Version#> as the target application. Click the Play button to run the script.

```
var doc = app.ActiveDoc;
var firstPgf = doc.MainFlowInDoc.FirstTextFrameInFlow.FirstPgf;
var tl = new TextLoc(firstPgf, 0);
doc.AddText (tl, "Hello");
doc.AutoChangeBars = 1;
```

The script creates a text location using the new `TextLoc()` method. Assign the first page of the document as the text location. Using the `AddText()` method, add a sample text. Set the `AutoChangeBars` property to 1 to enable the change bar.

### Changing the table properties in a document

The following script changes the left indentation of all the tables in a FrameMaker body page by 1 inch. It also changes the width of the columns to 2 inches. Open a FrameMaker document that has tables with two columns before running the script.

```
var doc = app.ActiveDoc;
var flow = doc.MainFlowInDoc;
var tbl = 0;
var textItems = flow.GetText(Constants.FTI_TblAnchor);
for (var i = 0; i < textItems.len; i += 1)
{
    tbl = textItems[i].obj;
    tbl.TblLeftIndent = (2*72 * 65536);
    var tblColWidths = new Metrics (2 * 72 * 65536, 2 * 72 * 65536);
    tbl.TblColWidths = tblColWidths;
}
```

The script creates the document's main flow using the `MainFlowInDoc` method. Use the `GetText(FTI_TblAnchor)` method to list all tables in the main flow. Using a for loop, for each table, use the `TblLeftIndent()` method to change the left indentation.

Use the `TblColWidths()` method to change the width of the column. This method accepts the argument in a metric form. So, create the arguments as a metric object using the `new Metrics()` method.

**NOTE:** By convention, there are 72 points per inch. Multiply the inch value with 65536 to get the correct value. 1 inch is equal to  $1 * 72 * 65536$  points.

# Differences between scripts and FDK

If you are familiar with the FDK (FrameMaker Developer Kit) object model, this chapter will help you get quickly get familiarized with ESTK. Adobe FrameMaker scripts are modeled closely on the FrameMaker FDK. These scripts act as wrappers to the FDK and hide the complexity of using FDK functions from users.

This chapter documents the differences between using scripts and using FDK.

## Naming differences between scripts and FDK

FDK	Scripts
Every FDK object is identified by an object ID.	Every script entity is identified as an object.
Every property in FDK starts with <code>FP_</code> . For example, <code>FP_CharTag</code> .	Properties are used directly, in scripts. Remove the <code>FP_</code> prefix before using the properties in scripts. For example, <code>FP_CharTag</code> in FDK becomes just <code>CharTag</code> in scripts. Every property has its own data type. The properties can either be read-only or read-write. <i>NOTE: All FDK properties can be used in FrameMaker scripts.</i>
Every API in FDK starts with <code>F_API</code> methodname.	Methods are used directly in scripts. Remove the <code>F_API</code> prefix before using the methods in scripts. For example, <code>F_ApiSplitElement</code> becomes just <code>SplitElement</code> in scripts.
Every object in FDK starts with <code>FO_</code> . For example, <code>FO_Doc</code> .	Use objects directly in scripts. Remove the <code>FO_</code> prefix before using the objects in scripts. For example, <code>FO_Doc</code> in FDK becomes just <code>Doc</code> in scripts. <i>NOTE: FDK object <code>FO_Session</code> is mapped to <code>app</code> and not to <code>Session</code>. This <code>app</code> script object is readily available for all FrameMaker scripts to use.</i>
Every structure in FDK starts with <code>F_</code> and ends with a <code>T</code> . For example, <code>F_AttributeT</code> .	Structures are used directly in scripts. Remove the <code>F_</code> prefix and the ending <code>T</code> before using the structures in scripts. For example, <code>F_AttributeT</code> in FDK becomes just <code>Attribute</code> in scripts. Some FDK structures contain other FDK structures within themselves. In a script, the structures within a structure are treated as an object data type. See the <code>F_TextRangeT</code> structure in <b>Example 3</b> below.



FDK	Scripts
FDK uses data types such as <code>StringT</code> for strings and <code>IntT</code> , <code>UIntT</code> for numbers.	Scripts map the FDK data types as follows: <code>StringT</code> is mapped to Javascript <code>String</code> . <code>IntT</code> and <code>UIntT</code> are mapped to Javascript <code>Number</code> . <code>ObjHandleT</code> is mapped to Javascript <code>Object</code> .

The following examples clearly highlight the differences between scripts and the FDK.

### Example 1:

The `app` property is readily available to all FrameMaker scripts and maps to the `FO_Session` object in FDK.

In this example, you query the `ActiveDoc` property that returns the `Active Document Object (Doc)` in the current FrameMaker Session. The reference to this object is stored in the variable `doc`.

```
doc = app.ActiveDoc;
flow = doc.MainFlowInDoc;
sync = flow.FlowIsSynchronized;
```

### Explanation:

- Line 1 stores the reference to the Active Document Object.
- Line 2 creates an object called `flow` by querying a property of the `doc` object.
- Line 3 queries a property of the newly created `flow` object (created in line 2).

You can query each of the object references as a new object recursively, until you reach the desired object property:

```
sync = app.ActiveDoc.MainFlowInDoc.FlowIsSynchronized;
```

The same example in FDK is as follows:

```
F_ObjHandleT docId, flowId;
IntT sync;
docId = F_ApiGetId(0, FV_SessionId, FP_ActiveDoc);
flowId = F_ApiGetId(FV_SessionId, docId, FP_MainFlowInDoc);
sync = F_ApiGetInt(docId, flowId, FP_FlowIsSynchronized);
```

As discussed in the table above, Example 1 demonstrates how FDK properties [`FP_ActiveDoc`, `FP_MainFlowInDoc` and `FP_FlowIsSynchronized`] are mapped to script properties [`ActiveDoc`, `MainFlowInDoc` and `FlowIsSynchronized`].

### Example 2:

Example 2 builds on Example 1 and shows how methods are called in scripts.

Every object has some methods of its own. These methods can be invoked in the same way as properties are invoked.

In Example 1, you created the `doc` object. To invoke a method, use `doc.methodname()`. For example, `doc.UpdateVariables()`;

This code calls the `UpdateVariables` method of the `doc` object.

FDK users can relate these methods to FDK APIs. The same code in FDK becomes:

```
F_ApiUpdateVariables(FV_SessionId, docId);
```

Notice how the FDK API `F_ApiUpdateVariables` becomes the method name `UpdateVariables` in scripts, as mentioned in the table above.

### Global Methods:

Not every method is accessible through a specific object. There are some methods that are not called through any objects but are called directly. These methods are called Global Methods. Examples of Global Methods include `ChooseFile()`, `CallClent()` and `Sleep()`.

In addition to these methods, all FDK Variables and Macros are available as Global Methods. These Variables usually start with `FV_` in FDK.

Consider the following line of code for FDK:

```
F_ApiSetInt(FV_SessionId, docId, FP_Capitalization,
FV_CAPITAL_CASE_SMALL);
```

The same code can be written in scripts as follows:

```
doc.Capitalization = FV_CAPITAL_CASE_SMALL;
```

However, there is a separate class called `Constants` that can be used to access these Global Methods. The script code can also be written as:

```
doc.Capitalization = Constants.FV_CAPITAL_CASE_SMALL;
```

### Example 3:

Every FDK Structure such as `F_TextItemT`, and `F_TextLocT` are also mapped to objects in scripts. All the fields of these structures are mapped to the read or write properties of these objects.

For example, structure `F_PointT`:

```
typedef struct {
MetricT x,y; /* The coordinate pair */
} F_PointT;
```

is mapped to the `Point` object in scripting. The `Point` object has two properties `x` and `y`.

See another FDK structure called `F_TextRangeT`:

```
typedef struct {
F_TextLocT beg; /* The beginning of the range */
F_TextLocT end; /* The end of the range */
} F_TextRangeT;
```

This structure is mapped to the `TextRange` object with both the properties `beg` and `end` being of type `TextLoc`.

Notice that `F_TextLocT` itself is another structure.

**NOTE:** Structure objects do not have any methods associated with them.

The following example script fetches an Active Document and adds some text in the first paragraph of the first flow of this document.

```

doc = app.ActiveDoc;
flow = doc.MainFlowInDoc;
textFrame = flow.FirstTextFrameInFlow;
pgf = textFrame.FirstPgf;
textLoc = new TextLoc();
textLoc.obj = pgf;
textLoc.offset = 0;
doc.AddText(textLoc, 'Sample Text');

```

Notice how the `F_TextLoc` FDK structure maps to the `TextLoc` object in scripts.

Lines 1, 2, 3, 4 can be combined and written as:

```
pgf = app.ActiveDoc.MainFlowInDoc.FirstTextFrameInFlow.FirstPgf;
```

Lines 6, 7, 8 can be combined and written by passing parameters in the constructor of the `TextLoc` object as:

```
textLoc = new TextLoc(pgf, 0);
```

Scripts have advanced features such as notifications and commands.

## Notifications

Notifications is the internal mechanism through which a script registered for a particular event is run when the event is triggered.

As an example, consider the following script:

```

/*
Sample script to demonstrate how to use notification mechanism of
FrameMaker though ExtendScript
*/
/*
There are basically two parts to a notification script.
1. Registering for a desired notification. (This is done by calling
'Notification()' with desired notifications as argument.)
2. Implement a handler for these notifications in your script. ( Implement
'Notify()' in script)
*/
/*
Part 1. Registering for Notifications
Notification()' is a global function and is available readily in
scripting. (Similar to F_ApiNotification())
The Syntax for Notification() is: Notification(int, int,.....,int, bool);
it takes:
a.) list of notifications (FA_Note_PreOpenDoc, FA_Note_PostOpenDoc etc.)
and
b.) param bool: true = register for provided notification list, false =
unregister for provided notification list.
(This parameter is optional and by default 'true')
Note: for more details please see F_ApiNotification() in FDK reference

```

```

guide.
*/

/*
Notification(FA_Note_PreOpenDoc,true);
This is same as Notification(FA_Note_PreSaveDoc)
or you can register for more than one notification at once
Notification(FA_Note_PostOpenDoc,FA_Note_PreSaveDoc);
Here we register for the notification: FA_Note_PostOpenDoc - this
notification is
triggered just after a document is opened.
Notification(FA_Note_PostOpenDoc,true);
*/

/*
Part 2: handler
This can be done by implementing function 'Notify()' that are called by
FrameMaker when any desired even occurs.(Similar to F_ApiNotify())
Syntax for this callback function is:
@params
note: notification id to identify the event.
object: current document or book object which generated the event.
sparam: The string, if any, associated with the notification
(For example, if the notification is for an Open or Save operation, sparam
specifies the pathname of the affected file. If the notification is for
text entry, sparam specifies the text the user typed.)
iparam: The Integer, if any, associated with the notification. Probably
the fcode or bit flags.
Notify(note, doc, sparam, iparam)
Note: for more details please see F_ApiNotify() in FDK reference guide.
*/
function Notify(note, object, sparam, iparam)
{
switch (note) {
case FA_Note_PostOpenDoc:
alert("Opened document " + sparam);
break;
}
}

```

The line of code:

```
Notification(FA_Note_PostOpenDoc,true);
```

registers this script for the `FA_Note_PostOpenDoc` notification. This notification is triggered **just after** any document is opened in FrameMaker.

The lines of code:

```
function Notify(note, object, sparam, iparam)
{
switch (note) {
case FA_Note_PostOpenDoc:
```

```

    alert("Opened document " + sparam);
    break;
}

```

pops up an alert with the file name just after the file is opened.

## Menus and Commands

Scripting allows you to add custom menus and associate commands with these menus. In addition, you can implement your own handlers for commands in a similar way as the FDK.

**IMPORTANT:** Ensure that you do not include any hypertext command in your custom menus. Because the hypertext commands can only be executed in the View Only mode, and in the View Only mode, the custom menus are not visible.

The following script adds two menus and associated commands to the FrameMaker main menu.

```

/*
In this script, we add two menu items to FrameMaker Session and associate
two different commands to these menu items
*/
// Main menu bar
/* First we get the object associated with Main menubar in FrameMaker*/
mMenu = app.GetNamedMenu("!MakerMainMenu") ;
// F_ApiGetNamedMenu(FV_SessionId, FO_Menu, (ConStringT) "!MakerMainMenu")
/*
Defining new menu items
DefineAndAddMenu(name, label)
FDK equivalent : F_ApiDefineAndAddMenu(F_ObjHandleT toMenuId, StringT
name, StringT label)
*/
nMenu1 = mMenu.DefineAndAddMenu("Menu1", "Menu1");
nMenu2 = mMenu.DefineAndAddMenu("Menu2", "Menu2");
// Adding command to menus
/*
DefineAndAddCommand(cmdID, name, label, shortcut)
FDK equivalent: F_ApiDefineAndAddCommand(IntT cmdID, F_ObjHandleT
toMenuId, StringT name, StringT label, StringT shortcut)
cmdID is a integer that will be passed to Command() function defined in
your script. It should be unique to the script.
*/
nMenu1.DefineAndAddCommand(1, "cmd1", "command1", "");
nMenu2.DefineAndAddCommand(2, "cmd2", "command2", "");
// Another way for creating a command and adding it to menu items.
//cmd3 = DefineCommand(3, "cmd3", "cmd3", "");
// FDK equivalent : F_ApiDefineCommand(IntT cmdID, StringT name, StringT
label, StringT shortcut)
// add the command to different menu items
//nMenu1.AddCommandToMenu(cmd3);
// FDK equivalent : ApiAddCommandToMenu(F_ObjHandleT toMenuId,

```

```

F_ObjHandleT commandId)
//nMenu2.AddCommandToMenu(cmd3);
// must call if script has been run through ESTK, redundant otherwise.
UpdateMenus();
// Call back equivalent to F_ApiCommand(command)
function Command(cmd){
switch(cmd) {
case 1:
alert("Command1 executed");
break;
case 2:
alert("Command2 executed");
break;
}
}

```

Let us examine the script in detail:

- 1) The line: `mMenu = app.GetNamedMenu("!MakerMainMenu")` ; fetches the name of the FrameMaker main menu and stores it in the variable `mMenu`.

- 2) The lines:

```

nMenu1 = mMenu.DefineAndAddMenu("Menu1", "Menu1");
nMenu2 = mMenu.DefineAndAddMenu("Menu2", "Menu2");

```

*add two new menu entries Menu1 and Menu2, under the main menu. The name of the main menu is already stored in step 1.*

- 3) The commands:

```

nMenu1.DefineAndAddCommand(1, "cmd1", "command1", "");
nMenu2.DefineAndAddCommand(2, "cmd2", "command2", "");

```

*add two new menu entries called Command 1 and Command 2 under the menu entries Menu1 and Menu2, that were defined in step 2.*

- 4) Finally, the block of code:

```

function Command(cmd) {
case 1: alert("Command 1 executed");
break;
case 2: alert("command 2 executed");
break;
}

```

*assigns the relevant commands to the menu entries defined in step 3.*

*The number after the keyword case is the menu ID. Look at the command in step 3:*

```

nMenu1.DefineAndAddCommand(1, "cmd1", "command1", "");

```

*The first value within the parenthesis is the menuID. This value is passed to the case statement.*

*The system pops up the relevant alert when the corresponding menu entry is clicked.*

# Object Reference

## AFrame

Property name	Data Type	Description
AFrameIsCropped	bool	Indicates whether the anchored frame is cropped or not. Possible values are: 0 - (Default) Indicates the frame is not cropped. 1 - Indicates the frame is cropped.
AFrameIsFloating	bool	Indicates whether the anchored frame is set to be floatable or not. Possible values are: 0 - (Default) Indicates the frame is not floatable. 1 - Indicates the frame is floatable.
Alignment	number	Indicates how the anchored frame is aligned. Possible values are: <ul style="list-style-type: none"> <li>• <code>Constants.FV_ALIGN_LEFT (0)</code></li> <li><code>Constants.FV_ALIGN_CENTER (1)</code></li> <li><code>Constants.FV_ALIGN_RIGHT (2)</code></li> <li><code>Constants.FV_ALIGN_INSIDE (3)</code></li> <li><code>Constants.FV_ALIGN_OUTSIDE (4)</code></li> </ul>

Property name	Data Type	Description
AnchorType	number	<p>Indicates the location where the frame is anchored.</p> <p>Possible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_ANCHOR_INLINE (1)</li> <li>Constants.FV_ANCHOR_TOP (2)</li> <li>Constants.FV_ANCHOR_BELOW (3)</li> <li>Constants.FV_ANCHOR_BOTTOM (4)</li> <li>Constants.FV_ANCHOR_SUBCOL_LEFT (5)</li> <li>Constants.FV_ANCHOR_SUBCOL_RIGHT (6)</li> <li>Constants.FV_ANCHOR_SUBCOL_NEAREST (7)</li> <li>Constants.FV_ANCHOR_SUBCOL_FARTHEST (8)</li> <li>Constants.FV_ANCHOR_SUBCOL_INSIDE (9)</li> <li>Constants.FV_ANCHOR_SUBCOL_OUTSIDE (10)</li> <li>Constants.FV_ANCHOR_TEXTFRAME_LEFT (11)</li> <li>Constants.FV_ANCHOR_TEXTFRAME_RIGHT (12)</li> <li>Constants.FV_ANCHOR_TEXTFRAME_NEAREST (13)</li> <li>Constants.FV_ANCHOR_TEXTFRAME_FARTHEST (14)</li> <li>Constants.FV_ANCHOR_TEXTFRAME_INSIDE (15)</li> <li>Constants.FV_ANCHOR_TEXTFRAME_OUTSIDE (16)</li> <li>Constants.FV_ANCHOR_RUN_INTO_PARAGRAPH (17)</li> </ul>
BaselineOffset	number	Baseline Offset



Property name	Data Type	Description
Element	<i>Element</i>	If the anchored frame is in a structured flow in a FrameMaker document, Element indicates the element containing the anchored frame.
InTextFrame	<i>TextFrame</i>	Indicates the text frame in which the anchored frame is placed.
InTextObj	FMOBJECT	Indicates the column or text frame in which the anchored frame is placed.
NextAFrame	<i>AFrame</i>	Contains the next anchored frame in the text frame.
PrevAFrame	<i>AFrame</i>	Contains the previous anchored frame in the text frame.
SideOffset	number	Denotes the near side offset.
TextLoc	<i>TextLoc</i>	Contains the location of the anchor symbol.
FirstGraphicInFrame	FMOBJECT	Contains the first object in the anchored frame.
LastGraphicInFrame	FMOBJECT	Contains the last object in the anchored frame.
Angle	number	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	number	Denotes the arrowhead base angle in degrees.
ArrowLength	number	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	number	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is <code>False</code> .
ArrowScaleHead	number	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	number	Denotes the arrowhead tip angle in degrees.

Property name	Data Type	Description
ArrowType	number	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	number	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	<i>Color</i>	Denotes the spot color.
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	number	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	number	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	number	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	<i>Group</i>	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.

Property name	Data Type	Description
HeadArrow	number	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	number	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	number	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	number	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.

Property name	Data Type	Description
LocY	number	Distance of the object from the top of the parent frame (in inches) The permissible range is –216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FObject	Denotes the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Denotes the next selected graphic object in the document.
ObjectAttributes	<i>Strings</i>	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	number	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>

Property name	Data Type	Description
Pen	number	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.
Runaround	number	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	number	Denotes the width of the runaround gap, if the object is a runaround object.
TailArrow	number	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.
TintPercent	number	Specifies the tint percentage.
Unique	number	Indicates the graphic object's internal unique number.
Width	number	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.
FirstGraphicInFrame	FObject	Contains the first object in the anchored frame.
LastGraphicInFrame	FObject	Contains the last object in the anchored frame.

Property name	Data Type	Description
UserString	string	A string to which clients can store private data.

## AFrame methods

*Delete, GetProps, SetProps, ObjectValid.*

## Arc

Property name	Data Type	Description
DTheta	number	Denotes the arc angle length in degrees. The permissible range is $-360$ degrees to $+360$ degrees.
Theta	number	Specifies the start angle in degrees. The permissible range is $0$ degrees to $360$ degrees.
Angle	number	Specifies the angle of the object's rotation.
ArrowBaseAngle	number	Indicates the arrowhead base angle in degrees.
ArrowLength	number	Denotes the arrowhead length (always rounded down to the nearest $1/256$ point).
ArrowScaleHead	number	Set to $1$ if the arrowhead is scaled as the line width changes. Set to $0$ , otherwise.
ArrowScaleFactor	number	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest $1/16$ point). It is not used if the ArrowScaleHead property is set to <code>False</code> .
ArrowTipAngle	number	Specifies the arrowhead tip angle in degrees.

Property name	Data Type	Description
ArrowType	number	Denotes the Arrowhead style. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x01)</li> <li>Constants.FV_ARROW_HOLLOW (0x02)</li> <li>Constants.FV_ARROW_FILLED (0x03)</li> </ul>
BorderWidth	number	Specifies the border width in points. The permissible range is 0.015 point to 360 point.
Color	<i>Color</i>	Indicates the spot color.
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a Metrics object. The 0 <sup>th</sup> element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	number	Denotes the fill pattern. The patterns are denoted by numbers between 0 and 15. The following constants are provided for the fill patterns 0, 7 and 15: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FMObject	Specifies the frame containing the graphic object.
GraphicCantBeSelected	number	Set to 1 if the graphic object cannot be selected. Else, set to 0.
GraphicsSelected	number	Set to 1 if the graphic object is selected, else set to 0.

Property name	Data Type	Description
GroupParent	<i>Group</i>	Indicates the group to which the object belongs. Anchored and unanchored frames do not have this property.
HeadArrow	number	Set to 1 if line has an arrowhead, set to 0 otherwise.
Height	number	Specifies the height of the object in points. The permissible range is from 0.125 pt to 3600 pt.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	number	Indicates the type of line end. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	number	Specifies the distance of the object from the left side of the parent, in inches. The permissible range is from -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set the LocX property for anchored frames.



Property name	Data Type	Description
LocY	number	Specifies the distance of the object from the top of the parent frame, in inches. The permissible range is from –216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You CANNOT set the LocY property for anchored frames.
NextGraphicInDoc	FObject	Indicates the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Indicates the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Specifies the next selected graphic object in the document.
ObjectAttributes	<i>Strings</i>	Contains a list of strings, each string expressing an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	number	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>

Property name	Data Type	Description
Pen	number	Indicates the pen pattern, denoted by numbers between 0 and 7. Constants are provided for some of the fill patterns: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FMOBJECT	Specifies the previous graphic object in the frame.
PrevGraphicInGroup	FMOBJECT	Indicates the previous graphic object in the group.
Runaround	number	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	number	Denotes the width of the runaround gap, if the object is a runaround object.
TailArrow	number	Set to 1 if the arrowhead is enabled at the beginning of the line. Set to 0, otherwise.
TintPercent	number	Specifies the tint percentage.
Unique	number	Denotes the internal identifier of the graphic object.
Width	number	Specifies the width of the object in points. The permissible range is 0.125 point to 3600 point.

## Arc methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## AttrCondExpr

Property name	Data Type	Description
NextAttrCondExprInDoc	<i>AttrCondExpr</i>	Indicates the pointer to the next <code>AttrCondExpr</code> object in the document.
AttrCondExprStr	string	The <code>AttrCondExpr</code> expression in string format.
AttrCondExprIsActive	bool	Set to True if this expression is applied to the document. (Application will mean filtering the document according the constraints in the expression). Set to False, otherwise.

### AttrCondExpr methods

[ApplyAttributeExpression](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## Attribute

Property name	Data Type	Description
name	string	The name of the attribute.
values	<i>Strings</i>	The attribute value.
valflags	uint	The validation error flags. This property is read only.
allow	uint	Allow error as special case to suppress reporting by validation.

### Attribute methods

[Attribute](#).

## AttributeDef

Property name	Data Type	Description
name	string	Name of the attribute
required	number	Set to 1 if the attribute is required. Else, set to 0.
flags	uint	<p>The flag value determines whether an attribute is hidden, read-only or neither. The following are the values that an attribute can have:</p> <ul style="list-style-type: none"><li>• Constants.FV_AF_READ_ONLY (0x0001) - The attribute value is read-only.</li></ul> <p><i>Constants.FV_AF_HIDDEN (0x0002) - The attribute value is hidden.</i></p> <p><i>NULL - The attribute value is neither read-only nor hidden.</i></p>

Property name	Data Type	Description
attrType	number	<p>Denotes the attribute value's type. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_AT_STRING (0) - Any arbitrary text string.</li> </ul> <p><i>Constants.FV_AT_STRINGS (1) - One or more arbitrary text strings.</i></p> <p><i>Constants.FV_AT_CHOICES (2) - A value from a list of choices.</i></p> <p><i>Constants.FV_AT_INTEGER (3) - A signed whole number (optionally restricted to a range of values).</i></p> <p><i>Constants.FV_AT_INTEGERS (4) - One or more integers (optionally restricted to a range of values).</i></p> <p><i>Constants.FV_AT_REAL (5) - A real number (optionally restricted to a range of values).</i></p> <p><i>Constants.FV_AT_REALS (6) - One or more real numbers (optionally restricted to a range of values).</i></p> <p><i>Constants.FV_AT_UNIQUE_ID (7) - A string that uniquely identifies an element.</i></p> <p><i>Constants.FV_AT_UNIQUE_IDREF (8) - A reference to a UniqueID attribute.</i></p> <p><i>Constants.FV_AT_UNIQUE_IDREFS (9) - One or more references to a UniqueID attribute.</i></p>
choices	<i>Strings</i>	The choices from which to select, if the Type parameter is set to Constants.FV_AT_CHOICES (2).
defValues	<i>Strings</i>	Default value of the attribute.
rangeMin	string	The minimum permissible value, if any.
rangeMax	string	The maximum permissible value, if any.

## AttributeDef methods

[AttributeDef](#).

## AttributeDefs

An Array of [AttributeDef](#) objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## AttributeDefs methods

[concat](#), [pop](#), [push](#).

## AttributeEx

Property name	Data Type	Description
name	string	The name of the attribute.
values	<a href="#">Strings</a>	The attribute value.
valflags	uint	The validation error flags. This property is read only.
allow	uint	Allow error as special case to suppress reporting by validation.

## Attributes

An Array of [Attribute](#) objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array.

## Attributes methods

[concat](#), [pop](#), [push](#).

## AttributesEx

An Array of [AttributeEx](#) objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array.

## AttributesEx methods

[concat](#), [pop](#), [push](#).

## BodyPage

Property name	Data Type	Description
MasterPage	string	Denotes the name of the master page background for the body page if the PageBackground property is set to Constants.FV_BGD_OTHER (0x02). The value is null if the PageBackgroundproperty is set to either Constants.FV_BGD_DEFAULT (0x00) or Constants.FV_BGD_NONE (0x01).

Property name	Data Type	Description
PageBackground	number	<p>Indicates the type of master page background. The possible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_BGD_DEFAULT (0x00): The page has a Left or Right master page background if the document is double-sided, or a Right master page background if the document is single-sided.</li> </ul> <p><i>Constants.FV_BGD_NONE (0x01): The page has no master page background.</i></p> <p><i>Constants.FV_BGD_OTHER (0x02): The page has the custom master page background specified by the MasterPage property.</i></p> <p>To determine whether a body page has a Left or a Right master page background, when its PageBackground property is set to Constants.FV_BGD_DEFAULT (0x00), query its PageIsRecto property.</p>
PageFrame	<i>UnanchoredFrame</i>	Denotes the page frame for the page.
PageHeight	number	Indicates the height of the page.
PageIsRecto	number	Set to 1 if the page is a Right page; Set to 0 if the page is a Left page.
PageNext	<i>BodyPage</i>	Denotes the next body page in the document.
PageNum	number	Contains the current page number.
PageNumString	string	Contains the page number string.
PagePrev	<i>BodyPage</i>	Denotes the previous body page in the document.
PageWidth	number	Denotes the width of the page.
PointPageNum	number	Indicates the point page number.



## BodyPage methods

[ApplyPageLayout](#), [Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## Book

Property name	Data Type	Description
BookDontUpdateReferences	number	Set to 0 if FrameMaker is set to update cross-references when it opens the book; Set to 1, otherwise.
BooksModified	number	Set to 1 if the book has been modified; Set to 0, otherwise.
BooksSelected	number	Set to 1 if the book icon in the book window is selected; Set to 0 otherwise.
FirstComponentInBook	<i>BookComponent</i>	Denotes the first component in the book.
FirstSelectedComponentInBook	<i>BookComponent</i>	Denotes the first selected component in the book.
IsIconified	number	Set to 1 if the book window is iconified; set to 0 otherwise.
IsInFront	number	Set to 1 if the book window is in front of other windows in the FrameMaker product session; Set to 0 otherwise.
IsOnScreen	number	Set to 1 if the document is visible on the screen; Set to 0 otherwise. Note: The IsOnScreen property is always 1 for books, and setting it to 0 has no effect.
Label	string	Denotes the title in the book window title bar.
Name	string	Indicates the path of the book.
NextOpenBookInSession	<i>Book</i>	Denotes the next open book in FrameMaker session's list of open books.

Property name	Data Type	Description
StatusLine	string	String that appears in the book status bar. Querying this string returns the status string.
ScreenHeight	number	Denotes the height of the book window in pixels.
ScreenWidth	number	Denotes the width of the book window in pixels.
ScreenX	number	Specifies the offset of the book window in pixels from the left side of the screen (or the left of the FrameMaker product application window). If you set a value that results in the book window being off the screen, that value is ignored and the old value is retained.
ScreenY	number	Indicates the offset of the book window in pixels from the top of the screen (or the top of the FrameMaker product application window ). If you set a value that results in the book window being off the screen, that value is ignored and the old value is retained.
TypeOfDisplayText	number	Specifies the type of text snippet to display for each icon in the book window. The two possible values are: <ul style="list-style-type: none"> <li>Constants.FV_BK_FILENAME (1)- displays the book component's filename</li> </ul> <i>Constants.FV_BK_TEXT (2) - displays the first paragraph of the component's first flow.</i>
AcrobatBookmarkDisplayTags	number	Set to 1 if the IncludeParagraphTagsinBookmarkText property is set to 1 (the paragraph tag is added before the paragraph text in each bookmark); Set to 0 otherwise.

Property name	Data Type	Description
DocAcrobatColumnArticleThreads	number	Set to 1 if you want separate article threads for each column; set to 0 if you want separate article threads for each text frame. <i>NOTE: The DocPDFNoArticleThread property must be set to FALSE, when setting this option.</i>
DocAcrobatDefaultsChanged	number	Set to 1 if the default heuristics for determining the paragraph level are disabled; Set to 0, otherwise.
DocAcrobatElementList	<i>Strings</i>	Contains a list of the element tags and context labels to include in bookmarks. <i>NOTE: The DocPDFElementList property applies only to structured FrameMaker documents.</i>
DocAcrobatElements	number	Set to 1 if elements rather than paragraphs are used for bookmarks; Set to 0 otherwise. <i>NOTE: The DocPDFElements property applies only to structured FrameMaker documents.</i>
DocAcrobatNoArticleThreads	number	Set to 1 if you do not want article threads in the resulting PDF; Set to 0 otherwise.
GenerateAcrobatInfo	number	Set to 1 if the GenerateAdobeAcrobatData property is set to True. Else, set to 0. To generate PDF data, you must set the other document print properties as follows: <ul style="list-style-type: none"> <li>PrintToFile - True</li> <li>PrintThumbnails - False</li> <li>PrintSeps - False</li> <li>PrintBlankPages - True</li> <li>PrintLastSheetFirst - False</li> <li>PrintNumCopies - 1</li> <li>PrintOddPages - True</li> <li>PrintEvenPages - True</li> <li>PrintScale - 100%</li> </ul>

Property name	Data Type	Description
PDFBookmark	bool	Set to True if FrameMaker is set to generate bookmarks when saving as PDF. Set to False, otherwise.
PDFBookmarksOpenLevel	number	Denotes the level of bookmarks to be expanded when Acrobat opens the generated PDF document. The value can be any integer, or one of the following constants: <ul style="list-style-type: none"> <li>Constants.FV_PDFBookmarksOpenDefaultLevel (-1)</li> <li>Constants.FV_PDFBookmarksOpenAllLevels (-2)</li> <li>Constants.FV_PDFBookmarksOpenNoneLevel (-3)</li> </ul> If you specify an integer that is greater than the number of levels in the Bookmarks Settings, the Constants.PDFBookmarksOpenAllLevels (-2) value takes effect.
PDFConvertCMYKtoRGB	number	When set to 1, the PDFConvertCMYKtoRGB property corresponds with the setting "Convert CMYK colors to RGB" in the Save As PDF dialog box.
PDFDestsMarked	number	Set to 1 if the document has paragraphs or elements marked through the MarkedForNamedDestination property. One of two things must occur in order for this property to be True: <ol style="list-style-type: none"> <li>1. The document must have been created and opened in FrameMaker 6.0 or later.</li> <li>2. The PDF FileSize Optimization client must have been run over it to mark all paragraphs or elements that are targets of hypertext links.</li> </ol>

Property name	Data Type	Description
PDFDistillerAbsent	number	A value of 1 indicates that there is no Acrobat Distiller available. Any other value indicates that Acrobat Distiller is available.
PDFDocInfo	<i>Strings</i>	Contains a list of strings expressing values to be set in the PDF Document Info dictionary when you save the book as PDF. Each dictionary entry is expressed as a pair of strings; the first string expresses the field name, and the second string expresses the field value.
PDFEndPage	string	Denotes the last page in the range of pages to be printed, in the FrameMaker numbering style.
PDFJobOption	string	Specifies the name of the Distiller Job Options. If the specified name does not exist in the Distiller Job Options list, then the first Distiller Job Option in the list is used.
PDFJobOptionsAbsent	number	A value of 1 indicates that PDF Job Options are not available. Any other value indicates that PDF Job Options are available.
PDFOpenPage	string	Denotes the PDF page number, in the FrameMaker numbering style, at which Acrobat opens the generated PDF document.
PDFPageHeight	number	Denotes the page height for the generated PDF document.
PDFPageWidth	number	Indicates the page width for the generated PDF document.
PDFPrintPageRange	number	Specifies the page range for generating the PDF document. If set, the PDF file contains only the specified pages. Else, FrameMaker generates PDF for the entire document or book.

Property name	Data Type	Description
PDFRegistrationMarks	number	<p>Contains the registration marks for the generated PDF. The permissible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_PDFRegistrationMarksNone (0)</li> <li>Constants.FV_PDFRegistrationMarksWestern (1)</li> <li>Constants.FV_PDFRegistrationMarksTombo (2)</li> </ul>
PDFSeparateFiles	number	<p>Set to 1, if a separate PDF file should be generated for each document in a book; Set to 0 otherwise. This property can be set for single documents, but is ignored in that case.</p>
PDFStartPage	string	<p>Denotes the first page of the printing page range, in the FrameMaker numbering style.</p>
PDFZoomFactor	number	<p>When the PDFZoomType property is set to Constants.FV_PDFZoomNone (0), the PDFZoomFactor property denotes the zoom percentage of the PDF document (metric 25% to 1600%). If the value is negative or zero, the value of Constants.FV_PDFZoomDefault (1) takes effect.</p>
PDFZoomType	number	<p>Denotes the PDF zoom setting at which Acrobat opens the generated PDF document. Permissible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_PDFZoomNone (0)</li> <li>Constants.FV_PDFZoomDefault (1)</li> <li>Constants.FV_PDFZoomPage (2)</li> <li>Constants.FV_PDFZoomWidth (3)</li> <li>Constants.FV_PDFZoomHeight (4)</li> </ul> <p>If a value other than these is specified, Constants.FV_PDFZoomDefault (1) takes effect.</p>

Property name	Data Type	Description
PDFAllNamedDestinations	number	Set to 1 if the generated PDF document should include Named Destinations for every paragraph and FrameMaker structure element in the book. This results in a larger PDF filesize. If set to 0, the generated PDF document will have Named Destinations only for those paragraphs and objects that have already been marked with PDFDestsMarked set to True.
PrintBlankPages	number	Set to 1 if the PageRounding property is set to allow empty pages at the end of documents; set to 0 otherwise.
PrintCollated	number	Set to 1 if Collate is enabled; Set to 0 otherwise.
PrintEmulsion	number	Denotes the direction of the print emulsion. The possible value are: <ul style="list-style-type: none"> <li>Constants.FV_EMUL_UP (0) - Emulsion side up</li> <li>Constants.FV_EMUL_DOWN (1) - Emulsion side down</li> </ul>
PrinterName	string	This property has no effect.
PrintEvenPages	number	Set to 1 if Print Even-Numbered Pages is enabled.
PrintFileName	string	Specifies the filename to print to. You can set the filename to be the default filename by specifying the PrintFileName value as null.
PrintImaging	number	Denotes the type of print imaging. The permissible values are: <ul style="list-style-type: none"> <li>Constants.FV_IMG_POSITIVE (0)</li> <li>Constants.FV_IMG_NEGATIVE (1)</li> </ul>
PrintLastSheetFirst	number	Set to 1 to print the last sheet first. Else, set to 0.
PrintLowRes	number	Set to 1 to print in low resolution. Else, set to 0.
PrintNumCopies	number	Denotes the number of copies to print.

Property name	Data Type	Description
PrintOddPages	number	Set to 1 if Print Odd-Numbered Pages is enabled.
PrintPaperHeight	number	Specifies the height of the printing paper.
PrintPaperWidth	number	Indicates the width of the printing paper.
PrintRegistrationMarks	number	Set to 1 if Registration Marks is enabled; Set to 0 otherwise.
PrintScale	number	Denotes the Print scale factor expressed as a percentage metric (0% to 100%). For print scale percentages, the value 1<<16 or 0x10000 specifies 100%.
PrintSeps	number	Set to 1 if Print Separations is enabled.
PrintToFile	number	Set to 1 if Print Only to File is enabled.
SkipBlankSeps	number	Set to 1 if Skip Blank Separations (do not print blank color separations) is enabled.
CustomElementList	<i>Strings</i>	Contains the list of tags to display when <code>ElementCatalogDisplay</code> is set to <code>Constants.FV_ELCAT_CUSTOM (0x04)</code> .
ElementCatalog	<i>ElementCatalogEntries</i>	Contains a list of elements in the Element Catalog.



Property name	Data Type	Description
ElementCatalogDisplay	number	<p>Catalog display options. Show tags for:</p> <ul style="list-style-type: none"> <li>Constants.FV_ELCAT_STRICT (0x00): Valid children for working from start to finish.</li> </ul> <p><i>Constants.FV_ELCAT_LOOSE (0x01): Valid children for working in any order.</i></p> <p><i>Constants.FV_ELCAT_CHILDREN (0x02): Children allowed anywhere in parent.</i></p> <p><i>Constants.FV_ELCAT_ALL (0x03): All elements.</i></p> <p><i>Constants.FV_ELCAT_CUSTOM (0x04): List of tags specified by the CustomElementList property.</i></p>
FileExtensionOverride	string	Denotes the filename extension to use when saving the document as XML. Typically, this property is used to save XHTML with a .htm extension rather than .xml. This setting should be set in the structure application for the document's DOCTYPE.
ElementSelection	<i>ElementRange</i>	Indicates the currently selected element range in the book.
FirstFmtChangeListInDoc	<i>FmtChangeList</i>	Specifies the ID of the first format change list in the list of format change lists in the book.
FirstElementDefInDoc	<i>ElementDef</i>	Denotes the first element definition in the list of element definitions in the document (FO_ElementDef ID).
HighestLevelElement	<i>Element</i>	Denotes the top most element for a structured book/document.

Property name	Data Type	Description
NewElemAttrDisplay	number	Specifies the attribute display properties for new elements. The permissible values are: <ul style="list-style-type: none"> <li>Constants.FV_ATTR_DISP_NONE (1): Do not display attributes.</li> <li>Constants.FV_ATTR_DISP_REQSPEC (2): Display required and specified attributes.</li> <li>Constants.FV_ATTR_DISP_ALL (3): Display all attributes.</li> </ul>
NewElemAttrEditing	number	Specifies when the Edit Attributes dialog box appears for new elements. The permissible values are: <ul style="list-style-type: none"> <li>Constants.FV_ATTR_EDIT_NONE (0)</li> <li>Constants.FV_ATTR_EDIT_REQUIRED (1)</li> <li>Constants.FV_ATTR_EDIT_ALWAYS (2)</li> </ul>
SeparateInclusions	number	Set to 1 if inclusions are listed separately in the element catalog; set to 0 otherwise.
StructuredApplication	string	Contains the name of the structured application that is associated with the book. If the book has no associated structure application, <code>GetString()</code> returns an empty string.
UseInitialStructure	number	Set to 1 if FrameMaker inserts the initial structure for new elements; set to 0 otherwise.
XmlDocType	string	Contains the DOCTYPE parameter from the source XML.
XmlEncoding	string	Specifies the encoding parameter of the XML Declaration for the source XML. The string is empty if no encoding is specified. If this property is set, the XML Declaration contains the encoding parameter with this value on Save As XML.

Property name	Data Type	Description
XmlFileEncoding	string	Specifies the encoding that is detected for the source XML book. If no encoding is specified for the source XML, XmlEncoding is an empty string. In that case, if this value is set, it determines the encoding to use when saving as XML. If XmlEncoding has a value, this string may be empty.
XmlPublicId	string	Contains the DOCTYPE public identifier for the source XML document.
XmlStandAlone	number	<p>Indicates an integer that specifies the XML standalone parameter for the XML document, that is the source of the current document. The permissible values are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_XML_STANDALONE_YES (1)</li> <li>  <i>Constants.FV_XML_STANDALONE_NO (2)</i></li> <li>  <i>Constants.FV_XML_STANDALONE_NONE (3)</i></li> <li>  <i>Constants.FV_XML_STANDALONE_NODEC (4)</i></li> </ul> <p>The standalone parameter is declared in the XML Declaration. For a file with no XML declaration, the value is Constants.FV_XML_STANDALONE_NODEC (4). For an XML declaration with no standalone parameter, this value is Constants.FV_XML_STANDALONE_NONE (3).</p>

Property name	Data Type	Description
XmlStyleSheet	string	<p>Specifies the XML stylesheet processing instruction to write out to XML when saving the book as XML.</p> <p>Note; The ESTK does not verify that you are using the correct syntax in this string.</p> <p>The string you set should not include the PI delimiters, &lt;? and ?&gt;. For example, the string you supply for my.css may be:</p> <pre> type="text\\css" href="my.css" </pre> <p>Only use this string to set a specific stylesheet specification. GetString() always returns null for this parameter. To get the list of stylesheet specifications associated with a book, use XmlStyleSheetList.</p>
XmlStyleSheetList	<i>Strings</i>	<p>Contains a list of stylesheet processing instructions for the current book. One book can have more than one stylesheet specification associated with it.</p> <p>Note: The ESTK does not verify that you are using the correct syntax in these strings.</p> <p>The strings should not include the PI delimiters, &lt;? and ?&gt;. For example, the string you supply for my.css may be:</p> <pre> type="text\\css" href="my.css" </pre> <p>Setting a list as the value to the XmlStyleSheetList property completely overwrites the preceding list.</p>
XmlSystemId	string	<p>Contains the DOCTYPE system identifier for the source XML document.</p>

Property name	Data Type	Description
XmlUseBOM	number	<p>Indicates whether a byte order mark was detected when opening the source XML. The permissible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_XML_USEBOM_YES (1) <i>Constants.FV_XML_USEBOM_NO</i> (2) <i>Constants.FV_XML_USEBOM_UTF8</i> (3) <i>Constants.FV_XML_USEBOM_UTF16BE</i> (4) <i>Constants.FV_XML_USEBOM_UTF16LE</i> (5) <i>Constants.FV_XML_USEBOM_UTF32BE</i> (6) <i>Constants.FV_XML_USEBOM_UTF32LE</i> (7)</li> </ul> <p>When saving as XML, if this property is set to Constants.FV_XML_USEBOM_YES (1), FrameMaker writes a byte order mark in the resulting XML.</p>
XmlVersion	string	<p>Depicts the XML Version that is specified in the XML Declaration when the file is opened. If no XML version is specified, <code>GetString()</code> returns an empty string.</p> <p>If this string contains an invalid XML declaration, a parsing error occurs when the book is saved as XML.</p>
XmlWellFormed	number	<p>Indicates whether the source XML is qualified as well formed. The permissible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_XML_WELLFORMED_YES (1) <i>Constants.FV_XML_WELLFORMED_NO</i> (2)</li> </ul>
BooksViewOnly	number	Set to 1 if the book is view-only; set to 0 otherwise.

Property name	Data Type	Description
ViewOnlyDeadCodes	<i>UInts</i>	Contains a list of F-codes that cannot be executed in the book.
ViewOnlyWinBorders	number	Set to 1 if the book has normal window borders; Set to 0 if the book's border buttons are suppressed.
ViewOnlyWinPopup	number	Set to 1 if the book window pop-up menu is available; set to 0 otherwise.
PDFGenerateForReview	number	This property is used to generate a PDF for review workflow that is getting review comments in PDF and importing them back to FrameMaker document. The property helps in importing back the comments correctly. When this flag is turned ON, the behavior is same as if the "Generate for review only" checkbox is ON in the PDF setup dialog.
ShowElementDescriptiveNames	bool	Set to True to show descriptive names for elements; Set to False, otherwise. <i>NOTE: The ShowElementDescriptiveNames property is applicable only for structured FrameMaker.</i>
UseInitialStructureOfAutoInsertedElements	bool	If set to True, auto-insertion rules will be processed recursively. For example: If an element is inserted automatically, and auto-insertion rules exist for this element, in the element definition, then those rules would also be processed.

## Book methods

*Close, Compare, GetProps, GetUniqueObject, Import, NewBookComponentInHierarchy, NewBookComponentOfTypeInHierarchy, NewSeriesBookComponent, NewSeriesObject, ObjectValid, Save, SetProps, SilentPrintDoc, SimpleGenerate, SimpleImportElementDefs, SimpleImportFormats, SimpleSave, Update-Book.*

## BookComponent

Property name	Data Type	Description
BookComponentIsGeneratable	number	Set to 1 if the book component is a generated file (BookComponentType is not set to Constants.FV_BK_NOT_GENERATABLE (16)); set to 0, otherwise.

Property name	Data Type	Description
BookComponentType	number	<p>Depicts the type of the book component. The possible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_BK_TOC (0) - Table of contents.</li> <li>Constants.FV_BK_LIST_FIGURE (1) - List of figures.</li> <li>Constants.FV_BK_LIST_TABLE (2) - List of tables.</li> <li>Constants.FV_BK_LIST_PGF (3) - List of paragraphs.</li> <li>Constants.FV_BK_LIST_MARKER (4) - List of markers.</li> <li>Constants.FV_BK_LIST_MARKER_ALPHA (5) - Alphabetical list of markers.</li> <li>Constants.FV_BK_LIST_PGF_ALPHA (6) - Alphabetical list of paragraphs.</li> <li>Constants.FV_BK_INDEX_STAN (7) - Standard index</li> <li>Constants.FV_BK_INDEX_AUTHOR (8) - Index of authors.</li> <li>Constants.FV_BK_INDEX_SUBJECT (9) - Subject index.</li> <li>Constants.FV_BK_INDEX_MARKER (10) - Index of markers.</li> <li>Constants.FV_BK_LIST_FORMATS (11) - List of formats</li> <li>Constants.FV_BK_LIST_REFERENCES (12) - List of references.</li> <li>Constants.FV_BK_INDEX_FORMATS (13) - Index of formats.</li> <li>Constants.FV_BK_INDEX_REFERENCES (14) - Index of references.</li> <li>Constants.FV_BK_NOT_GENERATABLE (16) - Book component is not a generated file.</li> </ul>
BookParent	<i>Book</i>	Specifies the book that contains the component (Constants.FO_BookID).



Property name	Data Type	Description
ChapNumComputeMethod	number	<p>The component document's chapter numbering type. The possible values are:</p> <ul style="list-style-type: none"> <li>• <i>Constants.FV_NUM_READ_FROM_FILE (0x00)</i> - Use the numbering properties from the document associated with this book component.</li> </ul> <p><i>Constants.FV_NUM_CONTINUE (0x01)</i> - Continue numbering from the previous chapter.</p> <p><i>Constants.FV_NUM_RESTART (0x02)</i> - Use the value specified for the <i>ChapterNumber</i> property.</p> <p><i>Constants.FV_NUM_SAME (0x03)</i> - Use the same chapter number as for the previous file.</p>
ChapterNumber	number	<p>If the <i>ChapNumComputeMethod</i> property is set to <i>Constants.FV_NUM_RESTART (0x02)</i>, use the <i>ChapterNumber</i> property as the chapter number.</p>

Property name	Data Type	Description
ChapterNumStyle	number	<p>Denotes the chapter numbering style. The possible value are:</p> <ul style="list-style-type: none"> <li>           Constants.FV_NUMSTYLE_NUMERIC (0x00) - Arabic  <i>Constants.FV_NUMSTYLE_ROMAN_UC (0x01) - Roman, uppercase</i>  <i>Constants.FV_NUMSTYLE_ROMAN_LC (0x02) - Roman, lowercase</i>  <i>Constants.FV_NUMSTYLE_ALPHA_UC (0x03) - Alphabetic, uppercase</i>  <i>Constants.FV_NUMSTYLE_ALPHA_LC (0x04) - Alphabetic, lowercase</i>  <i>Constants.FV_NUMSTYLE_KANJI (0x05) - Kanji</i>  <i>Constants.FV_NUMSTYLE_ZENKAKU (0x06) - Zenkaku</i>  <i>Constants.FV_NUMSTYLE_ZENKAKU_UC (0x07) - Zenkaku, uppercase</i>  <i>Constants.FV_NUMSTYLE_ZENKAKU_LC (0x08) - Zenkaku, lowercase</i>  <i>Constants.FV_NUMSTYLE_KANJI_KAZU (0x09) - Kazu</i>  <i>Constants.FV_NUMSTYLE_DAIJI (0x0a) - Daiji</i>  <i>Constants.FV_NUMSTYLE_TEXT (0x0b) - Text. This property is only for volume/chapter numbers.</i> </li> </ul>
ChapterNumText	string	<p>If the ChapNumStyle property is set to Constants.FV_NUMSTYLE_TEXT (0x0b), use the string specified in the ChapterNumText property as the chapter number.</p>

Property name	Data Type	Description
ComponentDisplayText	string	Specifies the text that displays in the book window when the value of the TypeOfDisplayText property is set to Constants.FV_BK_TEXT (2). Set the ComponentDisplayText property to an empty string ("") to allow FrameMaker to automatically update the text line with normal information.
ComponentIsSelected	number	Set to 1 if the component is selected in the book window; otherwise set to 0.
ExtractTags	<i>Strings</i>	List of paragraph tags or markers type names that are used to set up a generatable file (for example, table of contents, list of figures, standard index or index of authors).
FirstPageNum	number	Number for the first page in the component; used when the PageNumComputeMethod property is set to Constants.FV_NUM_RESTART (0x02).
FnFirstNum	string	Number for the first footnote in the component; used when the FnNumComputeMethod property is set to Constants.FV_NUM_RESTART (0x02).
FnCustNumString	string	Indicates the characters for custom document footnote numbers.

Property name	Data Type	Description
FnNumStyle	number	<p>Footnote numbering style. The possible values are:</p> <ul style="list-style-type: none"> <li>           Constants.FV_FN_NUM_NUMERIC (0x00) - Arabic  <i>Constants.FV_FN_NUM_ROMAN_UC (0x01) - Roman uppercase</i>  <i>Constants.FV_FN_NUM_ROMAN_LC (0x02) - Roman lowercase</i>  <i>Constants.FV_FN_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</i>  <i>Constants.FV_FN_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</i>  <i>Constants.FV_FN_NUM_CUSTOM (0x05) - Custom numbering</i>  <i>Constants.FV_FN_NUM_KANJI (0x06) - Kanji characters</i>  <i>Constants.FV_FN_NUM_ZENKAKU (0x07) - Zenkaku</i>  <i>Constants.FV_FN_NUM_ZENKAKU_UC (0x08) - Zenkaku uppercase</i>  <i>Constants.FV_FN_NUM_ZENKAKU_LC (0x09) - Zenkaku lowercase</i>  <i>Constants.FV_FN_NUM_KANJI_KAZU (0x0A) - Kazu</i>  <i>Constants.FV_FN_NUM_DAIJI (0x0B) - Daiji</i> </li> </ul>

Property name	Data Type	Description
FnNumComputeMethod	number	<p>The component document's footnote numbering type. The possible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_NUM_READ_FROM_FILE (0x00) - Use the numbering properties from the document associated with this book component.</li> </ul> <p><i>Constants.FV_NUM_CONTINUE (0x01) - Continue numbering from the previous file.</i></p> <p><i>Constants.FV_NUM_RESTART (0x02) - Use the number specified by the FnFirstNum property.</i></p> <p><i>Constants.FV_NUM_PER_PAGE (0x03) - Restart numbering on each page.</i></p>
GenerateInclude	number	Set to 1 if the document appears in the scroll list of files to be generated by the Generate/Update command for the book; otherwise set to 0.
ImportFmtInclude	number	Set to 1 if the book component is included in the list of components to be updated with imported formats or element definitions, when the user or a client executes Import Formats or Import Element Definitions; otherwise set to 0.
InsertLinks	number	Set to 1 if hypertext links are automatically inserted in generated files; otherwise set to 0.
Name	string	Indicates the path of the document that the component represents.
NextComponentInBook	<i>BookComponent</i>	Specifies the next component in the book file.
NextSelectedComponentInBook	<i>BookComponent</i>	Specifies the next selected component in the book window.

Property name	Data Type	Description
PageNumComputeMethod	number	<p>The component document's page numbering type. The possible values are:</p> <ul style="list-style-type: none"><li>• Constants.FV_NUM_READ_FROM_FILE (0x00) - Use the numbering properties from the document associated with this book component.</li></ul> <p><i>Constants.FV_NUM_CONTINUE (0x01) - Continue the numbering from the previous file.</i></p> <p><i>Constants.FV_NUM_RESTART (0x02) - Restart numbering at the value specified by the <i>FirstPageNum</i> property.</i></p>

Property name	Data Type	Description
PageNumStyle	number	<p>Page numbering style. The possible values are:</p> <ul style="list-style-type: none"> <li>           Constants.FV_PAGE_NUM_NUMERIC (0x00) - Arabic  <i>Constants.FV_PAGE_NUM_ROMAN_UC (0x01) - Roman uppercase</i>  <i>Constants.FV_PAGE_NUM_ROMAN_LC (0x02) - Roman lowercase</i>  <i>Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</i>  <i>Constants.FV_PAGE_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</i>  <i>Constants.FV_PAGE_NUM_KANJI (0x05) - Kanji characters</i>  <i>Constants.FV_PAGE_NUM_ZENKAKU (0x06) - Zenkaku</i>  <i>Constants.FV_PAGE_NUM_ZENKAKU_UC (0x07) - Zenkaku uppercase</i>  <i>Constants.FV_PAGE_NUM_ZENKAKU_LC (0x08) - Zenkaku lowercase</i>  <i>Constants.FV_PAGE_NUM_KANJI_KAZU (0x09) - Kazu</i>  <i>Constants.FV_PAGE_NUM_DAIJI (0x0a) - Daiji</i> </li> </ul>

Property name	Data Type	Description
PageSide	number	Specifies the page side on which to start the component document. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_BK_START_FROM_FILE (0x01)</li> <li>Constants.FV_BK_START_NEXT_AVAILABLE (0x02)</li> <li>Constants.FV_BK_START_LEFT (0x03)</li> <li>Constants.FV_BK_START_RIGHT (0x04)</li> </ul>
PgfNumComputeMethod	number	The type of numbering for the component document's paragraph. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_NUM_READ_FROM_FILE (0x00) - Use the numbering properties from the document associated with this book component.</li> <li>Constants.FV_NUM_CONTINUE (0x01) - Continue the numbering from the previous file.</li> <li>Constants.FV_NUM_RESTART (0x02) - Restart numbering at 1.</li> </ul>
PrevComponentInBook	<i>BookComponent</i>	Denotes the previous component in the book file.
PrintInclude	number	Set to 1 if the component document is included in the list of book files to be printed; otherwise set to 0.
TblFnCustNumString	string	Specifies the characters for custom table footnote numbers.



Property name	Data Type	Description
TbIFnNumStyle	number	<p>Table footnote numbering style. The possible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_FN_NUM_NUMERIC (0x00) - Arabic</li> <li>Constants.FV_FN_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>Constants.FV_FN_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>Constants.FV_FN_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>Constants.FV_FN_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>Constants.FV_FN_NUM_CUSTOM (0x05) - Custom numbering</li> <li>Constants.FV_FN_NUM_KANJI (0x06) - Kanji characters</li> <li>Constants.FV_FN_NUM_ZENKA-KU (0x07) - Zenkaku</li> <li>Constants.FV_FN_NUM_ZENKA-KU_UC (0x08) - Zenkaku uppercase</li> <li>Constants.FV_FN_NUM_ZENKA-KU_LC (0x09) - Zenkaku lowercase</li> <li>Constants.FV_FN_NUM_KANJI_KAZU (0x0A) - Kazu</li> <li>Constants.FV_FN_NUM_DAIJI (0x0B) - Daiji</li> </ul>
TbIFnNumComputeMethod	number	<p>The numbering type of the component document's table footnote. Possible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_NUM_READ_FROM_FILE (0x00) - Use the numbering properties from the document associated with this book component.</li> <li>Constants.FV_NUM_RESTART (0x02) - Start at 1.</li> </ul>

Property name	Data Type	Description
Unique	number	Represents the Unique ID of the book component.
VolNumComputeMethod	number	<p>The component document's volume numbering type. The possible values are:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_NUM_READ_FROM_FILE (0x00)</code> - Use the numbering properties from the document associated with this book component.</li> </ul> <p><i>Constants.FV_NUM_CONTINUE (0x01) - Continue numbering from the previous volume.</i></p> <p><i>Constants.FV_NUM_RESTART (0x02) - Use the value specified for the VolumeNumber property.</i></p> <p><i>Constants.FV_NUM_SAME (0x03) - Use the same volume number as for the previous file.</i></p>
VolumeNumber	number	<p>If the <code>VolNumComputeMethod</code> property is set to <code>Constants.FV_NUM_RESTART (0x02)</code>, use the value specified in the <code>VolumeNumber</code> property as the volume number.</p>

Property name	Data Type	Description
VolumeNumStyle	number	<p>The volume numbering style. The possible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_NUMSTYLE_NUMERIC (0x00) - Arabic</li> <li>Constants.FV_NUMSTYLE_ROMAN_UC (0x01) - Roman uppercase</li> <li>Constants.FV_NUMSTYLE_ROMAN_LC (0x02) - Roman lowercase</li> <li>Constants.FV_NUMSTYLE_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>Constants.FV_NUMSTYLE_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>Constants.FV_NUMSTYLE_KANJI (0x05) - Kanji</li> <li>Constants.FV_NUMSTYLE_ZENKAKU (0x06) - Zenkaku</li> <li>Constants.FV_NUMSTYLE_ZENKAKU_UC (0x07) - Zenkaku uppercase</li> <li>Constants.FV_NUMSTYLE_ZENKAKU_LC (0x08) - Zenkaku lowercase</li> <li>Constants.FV_NUMSTYLE_KANJI_KAZU (0x09) - Kazu</li> <li>Constants.FV_NUMSTYLE_DAIJI (0x0a) - Daiji</li> <li>Constants.FV_NUMSTYLE_TEXT (0x0b) - Text</li> </ul>
VolumeNumText	string	<p>If the <code>VolNumStyle</code> property is set to <code>Constants.FV_NUMSTYLE_TEXT (0x0b)</code>, use the string specified in the <code>VolumeNumText</code> property as the chapter number.</p>
ComponentElement	<i>Element</i>	<p>Specifies the component element (FO_Element ID).</p>

Property name	Data Type	Description
ExtractElementTags	<i>Strings</i>	List of element tags that are used to set up a generatable file (for example, table of contents, list of figures, or list of tables).
FirstComponentInBookComponent	<i>BookComponent</i>	Denotes the first component of the hierarchy for which the current component is the parent.
BookComponentParent	<i>BookComponent</i>	Denotes the parent of the current book element.
ExcludeBookComponent	number	Specifies the component to exclude when printing the book.
BookComponentTemplatePath	string	The path to the file that is used as a template for a folder.
BookComponentTitle	string	The name of a folder or a group component.
ComponentType	number	Indicates whether the component is a folder, group, or a document. The value is one of: <ul style="list-style-type: none"> <li>• <i>Constants.FV_BK_FOLDER (0x02)</i> - Folder</li> <li><i>Constants.FV_BK_GROUP (0x40)</i> - Group</li> <li><i>Constants.FV_BK_FM (0x8)</i> - FrameMaker document</li> <li><i>Constants.FV_BK_MIF (0x10)</i> - MIF document</li> <li><i>Constants.FV_BK_XML (0x20)</i> - XML document</li> </ul>
XmlApplicationForBookComponent	string	The application name for the XML that is set as template for the folder.

Property name	Data Type	Description
SecNumComputeMethod	number	<p>Indicates the type of numbering applied to a particular book component. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_NUM_CONTINUE (0x01) <i>Constants.FV_NUM_RESTART (0x02)</i> <i>Constants.FV_NUM_SAME (0x03)</i></li> </ul>
SectionNumStyle	number	<p>Denotes the section numbering style. Possible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_PAGE_NUM_NUMERIC (0x00) - Arabic <i>Constants.FV_PAGE_NUM_ROMAN_UC (0x01) - Roman uppercase</i> <i>Constants.FV_PAGE_NUM_ROMAN_LC (0x02) - Roman lowercase</i> <i>Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</i> <i>Constants.FV_PAGE_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</i> <i>Constants.FV_PAGE_NUM_KANJI (0x05) - Kanji characters</i> <i>Constants.FV_PAGE_NUM_ZENKAKU (0x06) - Zenkaku</i> <i>Constants.FV_PAGE_NUM_ZENKAKU_UC (0x07) - Zenkaku uppercase</i> <i>Constants.FV_PAGE_NUM_ZENKAKU_LC (0x08) - Zenkaku lowercase</i> <i>Constants.FV_PAGE_NUM_KANJI_KAZU (0x09) - Kazu</i> <i>Constants.FV_PAGE_NUM_DAIJI (0x0a) - Daiji</i></li> </ul>

Property name	Data Type	Description
SectionNumber	number	Denotes the section number of the specified component.
SectionNumText	string	The text to use for the SectionNumber property.
SubsecNumComputeMethod	number	Indicates the type of numbering applied to a particular book component. The value is one of: <ul style="list-style-type: none"><li>Constants.FV_NUM_CONTINUE (0x01)</li><li>Constants.FV_NUM_RESTART (0x02)</li><li>Constants.FV_NUM_SAME (0x03)</li></ul>

Property name	Data Type	Description
SubsectionNumStyle	number	<p>Denotes the subsection numbering style. Possible values are:</p> <ul style="list-style-type: none"> <li>Constants.FV_PAGE_NUM_NUMERIC (0x00) - Arabic</li> <li>Constants.FV_PAGE_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>Constants.FV_PAGE_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI (0x05) - Kanji characters</li> <li>Constants.FV_PAGE_NUM_ZENKAKU (0x06) - Zenkaku</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_UC (0x07) - Zenkaku uppercase</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_LC (0x08) - Zenkaku lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI_KAZU (0x09) - Kazu</li> <li>Constants.FV_PAGE_NUM_DAIJI (0x0a) - Daiji</li> </ul>
SubsectionNumber	number	Denotes the subsection number of the specified component.
SubsectionNumText	string	The text to use for the SubsectionNumber property.

Property name	Data Type	Description
NextBookComponentInDFSOrder	<i>BookComponent</i>	Denotes the pointer to traverse through book components in a hierarchal manner (top to bottom).
BookComponentFileType	number	Determines the file type of the document. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_BK_FM (0x8) - FrameMaker document</li> <li>Constants.FV_BK_MIF (0x10) - MIF document</li> <li>Constants.FV_BK_XML (0x20) - XML document</li> <li>Constants.FV_BK_XML (0x20) - Ditamap</li> </ul>
PrevBookComponentInDFSOrder	<i>BookComponent</i>	Denotes the pointer to traverse through book components in an heirarchial manner (bottom to top).
BookComponentIsFolderWithTemplate	number	Determines whether a template is associated with a book. Returns 1 if the <i>BookComponent</i> is a folder and a template is associated with the component. Returns 0, otherwise.
BookComponentIsFolderWithoutTemplate	number	Determines whether a template is not associated with a book. Returns 1 if the <i>BookComponent</i> is a folder and a template is not associated. Returns 0 otherwise.
ComponentIsDitaMap	bool	Determines whether a book component is a ditamap. Set to 1 if the book component is a ditamap; set to 0 otherwise.

## BookComponent methods

*Delete, GetProps, MoveComponent, ObjectValid, SetProps.*



# Cell

Property name	Data Type	Description
CellAboveInCol	<i>Cell</i>	Denotes the cell above the current cell (FO_CellID).
CellAngle	number	Indicates the angle of rotation of the cell.
CellBelowInCol	<i>Cell</i>	Denotes the cell below the current cell (FO_CellID).
CellColNum	number	Represents the cell's column number.
CellDefaultBottomRuling	<i>RulingFmt</i>	Indicates the cell's default bottom ruling (FO_RulingFmt ID).
CellDefaultLeftRuling	<i>RulingFmt</i>	Specifies the cell's default left ruling (FO_RulingFmt ID).
CellDefaultRightRuling	<i>RulingFmt</i>	Indicates the cell's default right ruling (FO_RulingFmt ID).
CellDefaultTopRuling	<i>RulingFmt</i>	Represents the cell's default top ruling (FO_RulingFmt ID).
CellsShown	number	Returns 1 if the cell is conditional and is visible; returns 0 otherwise.
CellsStraddled	number	Returns 1 if the cell is in a straddle but is not the first cell; returns 0 if the cell is the first cell in a straddle, or is not in a straddle.
CellNumColsStraddled	number	Specifies the number of columns in the straddle, if the cell is the first cell in a horizontal straddle; Otherwise the value is 1.
CellNumRowsStraddled	number	Specifies the number of rows in the straddle, if the cell is the first cell in a vertical straddle; Otherwise the value is 1.
ContentHeight	number	Denotes the distance between the top of the cell and the baseline of the last line in the cell.

Property name	Data Type	Description
CellOverrideBottomRuling	<i>RulingFmt</i>	Represents the cell's bottom ruling (FO_RulingFmt ID), if there is an override. The value is null if there is no override.
CellOverrideFill	number	Indicates the cell's fill pattern. The value is null if there is no override fill pattern.
CellOverrideLeftRuling	<i>RulingFmt</i>	Indicates the cell's left ruling (FO_RulingFmt ID), if there is an override. The value is null if there is no override.
CellOverrideRightRuling	<i>RulingFmt</i>	Specifies the cell's right ruling (FO_RulingFmt ID), if there is an override. The value is null if there is no override.
CellOverrideShading	<i>Color</i>	Denotes the cell's spot color (FO_Color ID). The value is null if there is no override shading.
CellOverrideTopRuling	<i>RulingFmt</i>	Denotes the cell's top ruling (FO_RulingFmt ID), if there is an override. The value is null if there is no override.
CellRow	<i>Row</i>	Denotes the row containing the cell (FO_RowID).
CellUseOverrideBRuling	number	Set to 1 if the cell's bottom ruling (specified by the <code>CellOverrideBottomRuling</code> property) overrides the default ruling specified by the table format; set to 0 otherwise.
CellUseOverrideFill	number	Set to 1 if the cell's fill pattern (specified by the <code>CellOverrideFill</code> property) overrides the default fill pattern specified by the table format; set to 0 otherwise.
CellUseOverrideLRuling	number	Set to 1 if the cell's left ruling (specified by the <code>CellOverrideLeftRuling</code> property) overrides the ruling specified by the table format; set to 0 otherwise.

Property name	Data Type	Description
CellUseOverrideRRuling	number	Set to 1 if the cell's right ruling (specified by the <code>CellOverrideRightRuling</code> property) overrides the ruling specified by the table format; set to 0 otherwise.
CellUseOverrideShading	number	Set to 1 if the cell's shading (specified by the <code>CellOverrideShading</code> property) overrides the default shading specified by the table format; set to 0 otherwise.
CellUseOverrideTRuling	number	Set to 1 if the cell's top ruling (specified by the <code>CellOverrideTopRuling</code> property) overrides the default top ruling specified by the table format; set to 0 otherwise.
Element	<i>Element</i>	Denotes the element containing the cell, if the cell is in a <code>FrameMaker</code> document.
FirstPgf	<i>Pgf</i>	Represents the first paragraph in the cell (FO_Pgf ID).
InTextFrame	<i>TextFrame</i>	Represents the text frame containing the cell (FO_TextFrame ID).
InTextObj	<i>SubCol</i>	Represents the text object containing the cell (FO_SubCol ID).
LastPgf	<i>Pgf</i>	Indicates the last paragraph in the cell (FO_Pgf ID).
NextCellInRow	<i>Cell</i>	Denotes the next cell in current row from left to right (FO_Cell ID).
NextCellInTbl	<i>Cell</i>	Represents the next cell from left to right (FO_Cell ID). If the cell is at the end of a row, the next cell is the first cell in the next row.
NextCell	<i>Cell</i>	Represents the next cell in the text frame (FO_Cell ID).

Property name	Data Type	Description
Overflowed	number	Specifies whether the text in the cell overflows. Set to 1 if the row Height Limit Maximum is too low to display all the text in the cell; set to 0 otherwise.
PrevCellInRow	<i>Cell</i>	Represents the previous cell in the current row (FO_Cell ID).
PrevCell	<i>Cell</i>	Denotes the previous cell in the text frame (FO_Cell ID).
Unique	number	Represents the cell's unique ID.
UserString	string	A string to which clients can store private data.

## Cell methods

[GetProps](#), [GetText](#), [ObjectValid](#), [SetProps](#), [StraddleCells](#), [UnStraddleCells](#).

## CharFmt

Property name	Data Type	Description
Capitalization	number	The capitalization type. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAPITAL_ - CASE_NORM (0) - Normal capitalization (mixed uppercase and lowercase)</li> <li>Constants.FV_CAPITAL_ - CASE_SMALL (1) - Small caps</li> <li>Constants.FV_CAPITAL_ - CASE_LOWER (2) - Lowercase letters only</li> <li>Constants.FV_CAPITAL_CASE_UPPER (3) - Uppercase letters only</li> </ul>
ChangeBar	number	Returns 1 if Change Bars are on.
CharTag	string	Denotes the character format's tag name.

Property name	Data Type	Description
Color	<i>Color</i>	Indicates the spot color ((FO_Color ID).
CombinedFontFamily	<i>CombinedFontDefn</i>	Denotes the combined font definition (FO_CombinedFontDefn).
FontAngle	number	Represents the font angle (specifies an index into the array of font angles provided by the session property FontAngleNames).
FontEncodingName	string	Indicates the encoding of the font.
FontFamily	number	Specifies the font family (specifies an index into the array of font families provided by the session property FontFamilyNames).
FontPlatformName	string	Name that uniquely identifies a font on a specific platform. For combined fonts, the name is the Asian font name.
FontPostScriptName	string	Name given to a font when it is sent to a PostScript printer. For combined fonts, the name is the Asian font name.
WesternFontPlatformName	string	Name that uniquely identifies the Roman component of a combined font on a specific platform.
WesternFontPostScriptName	string	Name given to the Roman component of a combined font when it is sent to a PostScript printer.
FontSize	number	Denotes the font size (2 point to 400 point).
FontVariation	number	Indicates the font variation (specifies an index into the array of font variations provided by the session property FontVariationNames).
FontWeight	number	Indicates the font weight (specifies an index into the array of font weights provided by the session property FontWeightNames).

Property name	Data Type	Description
Language	number	<p>Denotes the hyphenation and spell-checking language to use. The possible languages are:</p> <ul style="list-style-type: none"> <li>Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li>Constants.FV_LANG_ENGLISH (0x01)</li> <li>Constants.FV_LANG_BRITISH (0x02)</li> <li>Constants.FV_LANG_GERMAN (0x03)</li> <li>Constants.FV_LANG_SWISS_GERMAN (0x04)</li> <li>Constants.FV_LANG_FRENCH (0x05)</li> <li>Constants.FV_LANG_CANADIAN_FRENCH (0x06)</li> <li>Constants.FV_LANG_SPANISH (0x07)</li> <li>Constants.FV_LANG_CATALAN (0x08)</li> <li>Constants.FV_LANG_ITALIAN (0x09)</li> <li>Constants.FV_LANG_FINNISH (0x10)</li> <li>Constants.FV_LANG_SWEDISH (0x11)</li> <li>Constants.FV_LANG_JAPANESE (0x12)</li> <li>Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>Constants.FV_LANG_KOREAN (0x15)</li> <li>Constants.FV_LANG_PORTUGUESE (0x0A)</li> <li>Constants.FV_LANG_BRAZILIAN (0x0B)</li> <li>Constants.FV_LANG_DANISH (0x0C)</li> <li>Constants.FV_LANG_DUTCH (0x0D)</li> <li>Constants.FV_LANG_NORWEGIAN (0x0E)</li> </ul>

Property name	Data Type	Description
KernX	number	Horizontal kern value for manual kerning expressed as a percentage of an em (metric –1000% to 1000%). A positive value moves a character right and a negative value moves a character left.
KernY	number	Vertical kern value for manual kerning expressed as a percentage of an em (metric –1000% to 1000%). A positive value moves characters up and a negative value moves characters down.
Name	string	Specifies the character format's name.
NextCharFmtInDoc	<i>CharFmt</i>	Denotes the next character format in the document (FO_CharFmt ID).
Overline	number	Returns 1 if <i>Overline</i> is enabled; returns 0 otherwise.
PairKern	number	Returns 1 if <i>PairKern</i> is enabled; returns 0 otherwise.
Position	number	Denotes the vertical position of the character. Possible values are: <ul style="list-style-type: none"> <li>• <i>Constants.FV_POS_NORM</i> (0) - Normal</li> <li><i>Constants.FV_POS_SUPER</i> (1) - <i>Superscript</i></li> <li><i>Constants.FV_POS_SUB</i> (2) - <i>Subscript</i></li> </ul>
Stretch	number	Denotes the character stretch (set width) expressed as a percentage of normal stretch for the font (metric –10% to 1000%).
Strikethrough	number	Returns True if <i>Strikethrough</i> is enabled; otherwise returns False.
Underlining	number	Indicates the character tracking expressed as a percentage of an em (metric –100% to 1000%).

Property name	Data Type	Description
UseCapitalization	number	Returns 1 if the <code>Capitalization</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseChangeBar	number	Returns 1 if the <code>ChangeBar</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseColor	number	Returns 1 if the <code>Color</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseFontAngle	number	Returns 1 if the <code>FontAngle</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseFontFamily	number	Returns 1 if the <code>FontFamily</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseFontSize	number	Returns 1 if the <code>FontSize</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseFontVariation	number	Returns 1 if the <code>FontVariation</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseFontWeight	number	Returns 1 if the <code>FontWeight</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseKernX	number	Returns 1 if the <code>KernX</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseKernY	number	Returns 1 if the <code>KernY</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseOverline	number	Returns 1 if the <code>Overline</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UsePairKern	number	Returns 1 if the <code>PairKern</code> property overrides the default value; returns 0 if the "As Is" setting is used.



Property name	Data Type	Description
UsePosition	number	Returns 1 if the <code>Position</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UserString	string	A string to which clients can store private data.
UseStretch	number	Returns 1 if the <code>Stretch</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseStrikethrough	number	Returns 1 if the <code>Strikethrough</code> property overrides the default value; returns 0 if the "As Is" setting is used.
UseUnderlining	number	Returns 1 if the <code>Underlining</code> property overrides the default value; returns 0 if the "As Is" setting is used.
BkColor	FMOBJECT	Denotes the text background color. If this property is applied on a paragraph format, it denotes the background color of the whole paragraph; if it is applied on a character format, it denotes the background color of the text over which the character format is applied.
UseBkColor	bool	The <code>UseBkColor</code> is a Boolean flag which indicates whether the <code>BkColor</code> property is set or not. A value of 1 indicates the text has some background color. A value of 0 indicates a transparent background. <i>NOTE: When the <code>UseBkColor</code> property is set to 0, the <code>BkColor</code> value is not used.</i>

## CharFmt methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## CMSCheckinParam

Property name	Data Type	Description
version	int	Denotes the version of the checked-in document. This property uses values from the CMSVersioningStrategy enumeration: CMSSameVersion: Same version CMSMinorVersion: Minor version CMSMajorVersion: Major version
versionLabel	string	Specifies the label of the document version
description	string	Specifies the description of the document version
checkinComment	string	Specifies the check-in comments of the document version
makeThisCurrentVersion	bool	Determines whether the specified version is the current version. Set to True to make this checked-in document the current version; else set to False

## CMSDeleteParam

Property name	Data Type	Description
deleteAllVersion	bool	Determines whether to delete all versions of the current document. Set to True if user want to delete all the version of a file.; else set to False.
deleteAllDependents	bool	Determines whether to delete all dependents of the current document. True if user wants to delete all the dependents of a file; else set to False

## CMSInfo

Property name	Data Type	Description
cmsName	string	Specifies the name of the content management system.
userFields	string	Specifies the list of optional user fields. If no user fields is specified then its value is NULL.
userLoginUi	bool	Determines if the default Connection manager dialog is used for login. False if default Connection manager dialog is used for login; else set to True.

## CMSMenuItem

Property name	Data Type	Description
id	int	Specifies the ID of the menu item.
name	string	Specifies the name of the menu item.
flags	int	Specifies the type of menu. Uses the values from the following enumeration: FV_CMSMenu_Is_Item: Custom menu is single item FV_CMSMenu_Is_Disabled: Custom menu is disabled FV_CMSMenu_Is_Separator: Custom menu is separator FV_CMSMenu_Is_SubMenu: Custom menu is of type submenu

## CMSProperty

Property name	Data Type	Description
prop	int	Specifies the CMS Property Id. Use the value of enum F_CMSItemPropertyT (See following enum) or custom property added by the user (pending - peterb)
label	string	Specifies the label of the property
isMultiValue	bool	Determines if the property is multivalue. Set to True if the property is multivalue; else set to False.
isEditable	bool	Determines if the property is editable. Set to True if the property is editable; else set to False.
values	TypedVals	Specifies the values of the property

## CMSProperties

An Array of objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## CMSRegistration

Property name	Data Type	Description
CMSRegistration	object	CMSRegistration object created when user registers a new Content Management System using CMSRegister.

## CMSResult

Property name	Data Type	Description
status	StatusT	Specifies the Command status
opResult	int	Specifies the Operation result. If CMS Command needs CMSTree update, assign F_CMSOpResultT values (See following enum),else can return any value depending on operation. For example: opResult = True/False for FA_CMSIsValidCommand, opResult = CMSPropertyNewMaxOpCode for FA_CMSGetPropertyMaxOpCode. etc.
message	string	If operation fails, user can send error message to FrameMaker.For FA_CMSObjectOpenReadOnly, FA_CMSObjectEdit command, user can return file-name which is downloaded
cmsItems	CMSResult	List of CMS objects

## CMSSession

Property name	Data Type	Description
CMSSession	object	CMSSession object created when user creates a new content management session

## Color

Property name	Data Type	Description
Black	number	Represents the percentage of black (metric 0% to 100%).

Property name	Data Type	Description
ColorOverprint	number	Overprint setting for the color. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_COLOR_KNOCKOUT (0x00)</li> <li>Constants.FV_COLOR_OVERPRINT (0x01)</li> </ul>
ColorPrintCtl	number	Indicates the type of color printing used in the document. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_PRINT_SPOT (0x0)</li> <li>Constants.FV_PRINT_PROCESS (0x1)</li> <li>Constants.FV_PRINT_NO (0x2)</li> </ul>
ColorTintPercent	number	Specifies the percentage of the <code>TintBaseColor</code> to use for tinting. Possible values are the tint percentage (0% to 100%) or <code>Constants.FV_COLOR_NOT_TINTED</code> (0x7fffffff) if the color is not a tint.
ColorViewCtl	number	A 12-bit number for spot color views. The least significant 2 bits are View 1, the next 2 bits are View 2, and so on. The values of each 2-bit setting are one of the following: <ul style="list-style-type: none"> <li>Constants.FV_SEP_NORMAL (0x0)</li> <li>Constants.FV_SEP_NONE (0x1)</li> <li>Constants.FV_SEP_WHITE (0x2)</li> </ul>
Cyan	number	Indicates the percentage of cyan (metric 0% to 100%).
FamilyName	string	Color library name. Note that you must specify the full ink name, including any trademark symbols. For example, use "MUNSELL\xa8 Book of Color" for "MUNSELL® Book of Color."
InkName	string	Specifies the name of the color library pigment.
Magenta	number	Represents the percentage of magenta (metric 0% to 100%).

Property name	Data Type	Description
Name	string	Denotes the name of the color.
NextColorInDoc	<i>Color</i>	Indicates the next color in the document (FO_Color ID).
ReservedColor	number	Color names reserved by FrameMaker. The values are: <ul style="list-style-type: none"> <li>Constants.FV_COLOR_NOT_RESERVED (0)</li> <li>Constants.FV_COLOR_CYAN (1)</li> <li>Constants.FV_COLOR_MAGENTA (2)</li> <li>Constants.FV_COLOR_YELLOW (3)</li> <li>Constants.FV_COLOR_BLACK (4)</li> <li>Constants.FV_COLOR_WHITE (5)</li> <li>Constants.FV_COLOR_RED (6)</li> <li>Constants.FV_COLOR_GREEN (7)</li> <li>Constants.FV_COLOR_BLUE (8)</li> </ul>
TintBaseColor	<i>Color</i>	Denotes the color from which the tint is derived (FO_Color ID), or Constants.FV_NO_BASE_COLOR (0x00) if the color is not a tint.
UserString	string	A string to which clients can store private data.
Yellow	number	Percentage of yellow (metric 0% to 100%).

## Color methods

*Delete, GetProps, ObjectValid, SetProps.*

## Column

Property name	Data Type	Description
InCond	Objects	Condition tags for row (array of FO_CondFmt IDs).

Property name	Data Type	Description
PrevColumnInTbl	Column	Indicates the previous column in the table
NextColumnInTbl	Column	Indicates the next column in the table
PrevVisibleColumnInTbl	Column	Indicates the previous visible column in the table
NextVisibleColumnInTbl	Column	Indicates the next visible column in the table
ColumnsShown	int	Indicates the visible columns in the table
ColumnTbl	Tbl	Table containing the column

## Column methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## CombinedFont

Property name	Data Type	Description
combinedFont	<a href="#">CombinedFontDefn</a>	ID of the combined font.
variation	uint	Index of the font variation.
weight	uint	Index of the font weight.
angle	uint	Index of the font angle.

## CombinedFont methods

[CombinedFont](#).



## CombinedFontDefn

Property name	Data Type	Description
NextCombinedFontDefnInDoc	<i>CombinedFontDefn</i>	Represents the next combined font definition instance in the document (FO_CombinedFontDefn ID).
Name	string	Indicates the name of the combined font.
BaseFamily	number	Asian font family (specifies index into the arrays of font families provided by the session property, <code>FontFamilyNames</code> )
WesternFamily	number	Specifies the western font family (specifies index into the arrays of font families provided by the session property, <code>FontFamilyNames</code> ).
ViewHotspotIndicators	Bool	Turns on hotspot indicators. Hotspot indicators are small square boxes at the centre of an object to indicate that the object is actually a hotspot.
WesternSize	number	Represents the scaling factor for Roman text expressed as a percentage of base font size (metric 1% to 1000%).
WesternShift	number	Indicates the baseline offset of Roman text expressed as a percentage of base font size (metric 1% to 1000%).
FontEncodingName	string	Specifies the combined font's encoding, based on the BaseFamily property.
UserString	string	Represents a string that your client can set as the value for the <code>FO_CombinedFontDefn</code> object.

### CombinedFontDefn methods

*CombinedFamilyFonts, Delete, GetProps, ObjectValid, SetProps.*

## CombinedFonts

An Array of *CombinedFont* objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array.

### CombinedFonts methods

*concat*, *pop*, *push*.

## Command

Property name	Data Type	Description
CanHaveCheckMark	number	Returns True if the menu item can have a check mark. If the menu item is defined by FrameMaker, you can read this property, but not set it.
CheckMarkIsOn	number	Returns 1 if the menu item can have a check mark and the check mark is on. If the menu item is defined by FrameMaker, you can read this property, but not set it.
CommandNum	number	Represents the integer that you specified for the <code>cmd</code> parameter of <code>DefineAndAddCommand()</code> or <code>DefineCommand()</code> . When the user executes the command, FrameMaker passes this integer to your client's <code>Command()</code> function. If the menu item is defined by FrameMaker, you can read this property, but not set it.

Property name	Data Type	Description
EnabledWhen	number	<p>The context in which the menu item is enabled.. If the menu item is defined by FrameMaker, you can read this property, but not set it.</p> <p>The following are the values that the <code>EnabledWhen</code> property can have and the corresponding contexts in which a menu item is active.</p> <ul style="list-style-type: none"> <li> <p><code>Constants.FV_ENABLE_OBJ_PROPS (8)</code> - The insertion point is in text, a table, or a math object, or a graphic object is selected.</p> <p><i>Constants.FV_ENABLE_IN_TABLE (9)</i> - The insertion point or selection is in any part of a table.</p> <p><i>Constants.FV_ENABLE_IN_TABLE_TITLE (10)</i> - The insertion point or selection is in the table title.</p> <p><i>Constants.FV_ENABLE_IN_CELL_TEXT (11)</i> - The insertion point or selection is in a table cell.</p> <p><i>Constants.FV_ENABLE_IS_CELL (12)</i> - A single cell in a table is selected.</p> <p><i>Constants.FV_ENABLE_IS_CELLS (13)</i> - One or more cells in a table are selected.</p> </li> </ul>

Property name	Data Type	Description
		<ul style="list-style-type: none"> <li>• Constants.FV_ENABLE_IS_TABLE(14) - An entire table is selected.</li> <li><i>Constants.FV_ENABLE_IS_OBJ(15) - An object is selected.</i></li> <li><i>Constants.FV_ENABLE_IS_TEXT_FRAME (16) - A text frame is selected.</i></li> <li><i>Constants.FV_ENABLE_IS_OR_IN_FRAME (17) - The selected object is a graphic frame or is in a graphic frame that is not a page frame.</i></li> <li><i>Constants.FV_ENABLE_IS_AFRAME (18) - The first selected object is an anchored frame.</i></li> <li><i>Constants.FV_ENABLE_IS_TEXT_INSET (19) - The first selected object is a text inset.</i></li> <li><i>Constants.FV_ENABLE_IS_GRAPHIC_INSET (20) - The first selected object is a graphic inset.</i></li> <li><i>Constants.FV_ENABLE_IN_FLOW (21) - A text frame is selected, or the insertion point or selection is in a paragraph.</i></li> <li><i>Constants.FV_ENABLE_COPY(22) - Some text or an object is selected.</i></li> </ul>

Property name	Data Type	Description
		<ul style="list-style-type: none"> <li>Constants.FV_ENABLE_COPY_FONT (23) - The insertion point or selection is in the text of a paragraph, a math object, a table, or a text line.</li> </ul> <p><i>Constants.FV_ENABLE_CAN_PASTE(24) - The Clipboard contains an object or text that can be pasted at the insertion point.</i></p> <p><i>Constants.FV_ENABLE_IS_VIEW_ONLY (25) - The current document is locked.</i></p> <p><i>Constants.FV_ENABLE_NEEDS_DOCP_ONLY (26) - A document is open.</i></p> <p><i>Constants.FV_ENABLE_NEEDS_BOOKP_ONLY (27) - A book is open.</i></p> <p><i>Constants.FV_ENABLE_NEEDS_DOCP_OR_BOOKP (28) - A document or a book is open.</i></p>
ExpandOMaticParent	<i>Command</i>	Denotes the virtual parent object, if the menu item is an expandomatic menu item.
Fcode	uint	Specifies an f-code that FrameMaker executes when the user chooses the menu item or presses the keyboard shortcut.
Fcodes	<i>UInts</i>	The list of f-codes that the FrameMaker product executes when the user chooses the menu item or presses the keyboard shortcut. Normally, the first f-code in the list is the same as the f-code specified by the Fcode property.

Property name	Data Type	Description
HasShiftOrUnshiftCommand	number	<p>Specifies whether a command has an accompanying shift command or unshift command. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_ITEM_HAS_SHIFT_COMMAND (1)</li> <li>Constants.FV_ITEM_HAS_UNSHIFT_COMMAND (2)</li> <li>Constants.FV_ITEM_HAS_NO_SHIFT_OR_UNSHIFT_COMMAND (3)</li> </ul>
HelpLink	string	<p>The hypertext link to call when the user requests context-sensitive help for the command.</p> <p>If you set the HelpLink property, specify the destination file and an optional page number or linkname. For example, specify foo.doc:lastpage. Do not specify hypertext commands such asgotopage.</p> <p>FrameMaker automatically prefixes the appropriate hypertext command to the HelpLink string when the user requests context-sensitive help.</p> <p>If the destination file is not in the client directory, FrameMaker looks for the file in the FrameMaker product help directory.</p> <p><b>NOTE:</b> The HelpLink property is valid only for commands created by clients, and is not valid for commands created directly by FrameMaker.</p>
KeyboardShortcutLabel	string	<p>The keyboard shortcut string that appears on the menu. This string need not be one of the actual shortcuts specified by the KeyboardShortcuts property.</p>
KeyboardShortcuts	<i>Strings</i>	<p>The list of keyboard shortcuts that the user can press to execute the command. To add a shortcut, append it to the list.</p> <p><b>NOTE:</b> ESTK does not allow you to delete shortcuts from the list.</p>

Property name	Data Type	Description
Labels	<i>Strings</i>	<p>If the command is a menu item, the Labels property depicts the list of labels the menu item can have in different contexts. If the menu item has only one label in all contexts, Labels specifies only the string for that specific label. If the menu item has different labels in different contexts, Labels specifies pairs of strings with the following format:</p> <p><i>Context, Label</i></p> <p>where Label specifies the menu item label and Context specifies the context in which the label appears on the menu.</p>
MenuItemType	number	<p>The type of command or menu item. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_MENUITEM_FRAME (1) - the command is a menu item defined by FrameMaker.</li> </ul> <p><i>Constants.FV_MENUITEM_API (2) - the command is a menu item defined by a client.</i></p> <p><i>Constants.FV_MENUITEM_MACRO (3) - the menu item is not a command; it calls a macro.</i></p> <p><i>Constants.FV_MENUITEM_EXPANDOMATIC (4) - the menu item is an expandomatic menu item (such as !ShowParagraphTags) defined by FrameMaker.</i></p>
Mode	number	<p>The mode in which keyboard shortcuts are recorded. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_MODE_MATH (1)</li> <li><i>Constants.FV_MODE_NONMATH (2)</i></li> <li><i>Constants.FV_MODE_ALL (3)</i></li> </ul>
NextCommandInSession	<i>Command</i>	Represents the next command in the list of commands in the session.

Property name	Data Type	Description
ShiftOrUnshiftCommand	<i>Command</i>	If the <code>HasShiftOrUnshiftCommand</code> property is set to <code>Constants.FV_ITEM_HAS_SHIFT_COMMAND (1)</code> , the <code>ShiftOrUnshiftCommand</code> property, represents the command to use when the user holds down the Shift key. If the <code>HasShiftOrUnshiftCommand</code> property is set to <code>Constants.FV_ITEM_HAS_UNSHIFT_COMMAND (2)</code> , the <code>ShiftOrUnshiftCommand</code> denotes the command to use when the user is not holding down the Shift key.
Label	string	Indicates the dialog box title.
MenuItemsEnabled	number	Returns 1 if the menu or menu item is enabled. Returns 0 if the menu or menu item is disabled (dimmed).
Name	string	Represents the name of the command, menu, or menu item separator.
NextMenuItemInMenu	FObject	Indicates the next menu item, menu, or separator in the menu.
NextMenuItemInSession	FObject	Specifies the next menu item, menu, or separator in the list of menu items, menus, and separators in the session.
PrevMenuItemInMenu	FObject	Specifies the previous menu item, menu, or separator in the list of menu items, menus, and separators in the session.



Property name	Data Type	Description
Product	int	Specifies the Product to add the command. Values: Constants.FV_PRODUCT_UNSTRUCTURED Constants.FV_PRODUCT_STRUCTURED Constants.FV_PRODUCT_XMLAUTHOR Constants.FV_PRODUCT_ALL (Equivalent to: Constants.FV_PRODUCT_UNSTRUCTURED   Constants.FV_PRODUCT_STRUCTURED   Constants.FV_PRODUCT_XMLAUTHOR)
View	int	Specifies the View to add the command. Values: Constants.FV_VIEW_WYSIWYG Constants.FV_VIEW_AUTHOR Constants.FV_VIEW_XML Constants.FV_VIEW_ALL (Equivalent to: Constants.FV_VIEW_WYSIWYG   Constants.FV_VIEW_AUTHOR   Constants.FV_VIEW_XML)
Window	int	Specifies the Window to add the command. Values: Constants.FV_WINDOW_DOC Constants.FV_WINDOW_BOOK Constants.FV_WINDOW_DITAMAP Constants.FV_WINDOW_ALL (Equivalent to: Constants.FV_WINDOW_DOC   Constants.FV_WINDOW_BOOK   Constants.FV_WINDOW_DITAMAP)

## Command methods

*Delete, Delete, GetProps, ObjectValid, SetProps.*

## CompareRet

Property name	Data Type	Description
sum	<i>Doc</i>	Specifies the summary document.
comp	<i>Doc</i>	Specifies the composite document.

### CompareRet methods

*CompareRet.*

## CondFmt

Property name	Data Type	Description
CondFmtIsShown	number	Returns True if the condition is shown. To hide text with a specified condition, set this property and the <code>ShowAll</code> property of the <code>Doc</code> object to False.
Name	string	Denotes the name of the condition format.
NextCondFmtInDoc	<i>CondFmt</i>	Denotes the next condition format in the document (FO_CondFmt ID).
SepOverride	<i>Color</i>	Indicates the color separation format override (FO_Color ID).

Property name	Data Type	Description
StyleOverride	number	<p>The style condition indicators for conditional text. The possible values are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_CN_NO_OVERRIDE (0)  <i>Constants.FV_CN_OVERLINE (1)</i>  <i>Constants.FV_CN_STRIKE-THROUGH (2)</i>  <i>Constants.FV_CN_SINGLE_UNDERLINE (3)</i>  <i>Constants.FV_CN_DOUBLE_UNDERLINE (4)</i>  <i>Constants.FV_CN_CHANGEBAR (5)</i>  <i>Constants.FV_CN_NUMERIC_UNDERLINE (6)</i>  <i>Constants.FV_CN_NMRIC_AND_CHNGBAR (7)</i></li> </ul>
UseSepOverride	number	Returns 1 if color specified by the SepOverride property is used instead of the default color.
BkColor	<i>Color</i>	<p>BkColor is the text background color property whose value is the color to be used.</p> <p>If this property is applied on a paragraph format, the color is applied as the background color of the whole paragraph.</p> <p>If this property is applied on a character format, the color is applied as the background color of the text over which the character format is applied.</p>
UseBkColor	bool	<p>The UseBkColor is a Boolean flag which indicates whether the BkColor is set or not. UseBkColor=1 indicates the text has some background color. UseBkColor=0 indicates transparent background.</p> <p>When UseBkColor=0, the BkColor value is not used.</p>

Property name	Data Type	Description
UserString	string	A string to which clients can store private data.

## CondFmt methods

*Delete*, *GetProps*, *ObjectValid*, *SetProps*.

## Constants

**NOTE:** Adobe and Pantone have been working together to support your color needs. Pantone Matches are no longer supported in Adobe. Pantone color libraries currently preloaded in FrameMaker and FrameMaker Publishing Server will be phased out starting August 31, 2022 (with the exception of PANTONE+ CMYK Coated, PANTONE+ CMYK Uncoated, PANTONE+ Metallic Coated).

Property name	Data Type
Constants.FDK_LIBRARY_VERSION	int Value:0x17000000
Constants.FE_Success	int Value:0
Constants.FE_Transport	int Value:-1
Constants.FE_BadDocId	int Value:-2
Constants.FE_BadObjId	int Value:-3
Constants.FE_BadPropNum	int Value:-4
Constants.FE_BadPropType	int Value:-5
Constants.FE_ReadOnly	int Value:-6
Constants.FE_OutOfRange	int Value:-7
Constants.FE_DocModified	int Value:-8

Property name	Data Type
Constants.FE_GroupSelect	int <b>Value:-9</b>
Constants.FE_WithinFrame	int <b>Value:-10</b>
Constants.FE_NotGraphic	int <b>Value:-11</b>
Constants.FE_NotFrame	int <b>Value:-12</b>
Constants.FE_NotGroup	int <b>Value:-13</b>
Constants.FE_BadNewFrame	int <b>Value:-14</b>
Constants.FE_BadNewGroup	int <b>Value:-15</b>
Constants.FE_BadNewSibling	int <b>Value:-16</b>
Constants.FE_BadDelete	int <b>Value:-17</b>
Constants.FE_BadPageDelete	int <b>Value:-18</b>
Constants.FE_TypeUnNamed	int <b>Value:-19</b>
Constants.FE_NameNotFound	int <b>Value:-20</b>
Constants.FE_OffsetNotFound	int <b>Value:-21</b>
Constants.FE_SomeUnresolved	int <b>Value:-22</b>
Constants.FE_BadNew	int <b>Value:-23</b>
Constants.FE_NotBodyPage	int <b>Value:-24</b>
Constants.FE_NotPgf	int <b>Value:-25</b>

Property name	Data Type
Constants.FE_NotBookComponent	int Value:-26
Constants.FE_BadOperation	int Value:-27
Constants.FE_BadElementDefId	int Value:-28
Constants.FE_BadElementId	int Value:-29
Constants.FE_BadNotificationNum	int Value:-30
Constants.FE_BadContainer	int Value:-104
Constants.FE_BadTemplatePath	int Value:-105
Constants.FE_BadXmlApplication	int Value:-106
Constants.FE_DupName	int Value:-32
Constants.FE_BadName	int Value:-33
Constants.FE_CompareTypes	int Value:-34
Constants.FE_BadCompare	int Value:-35
Constants.FE_BadRange	int Value:-36
Constants.FE_PageFrame	int Value:-37
Constants.FE_CantSmooth	int Value:-38
Constants.FE_NotTextFrame	int Value:-39
Constants.FE_HiddenPage	int Value:-40

Property name	Data Type
Constants.FE_NotTextObject	int <b>Value:-41</b>
Constants.FE_SystemError	int <b>Value:-42</b>
Constants.FE_BadParameter	int <b>Value:-43</b>
Constants.FE_Canceled	int <b>Value:-44</b>
Constants.FE_FailedState	int <b>Value:-45</b>
Constants.FE_WantsCustom	int <b>Value:-46</b>
Constants.FE_WantsLandscape	int <b>Value:-47</b>
Constants.FE_WantsPortrait	int <b>Value:-48</b>
Constants.FE_ViewOnly	int <b>Value:-49</b>
Constants.FE_BadSaveFileName	int <b>Value:-50</b>
Constants.FE_GenRuleItemExpected	int <b>Value:-51</b>
Constants.FE_GenRuleMixedConnectors	int <b>Value:-52</b>
Constants.FE_GenRuleLeftBracketExpected	int <b>Value:-53</b>
Constants.FE_GenRuleRightBracketExpected	int <b>Value:-54</b>
Constants.FE_GenRuleAmbiguous	int <b>Value:-55</b>
Constants.FE_GenRuleSyntaxError	int <b>Value:-56</b>
Constants.FE_GenRuleConnectorExpected	int <b>Value:-57</b>

Property name	Data Type
Constants.FE_InvalidString	int <b>Value:-58</b>
Constants.FE_BadSelectionForOperation	int <b>Value:-59</b>
Constants.FE_WrongProduct	int <b>Value:-60</b>
Constants.FE_BookStructured	int <b>Value:-61</b>
Constants.FE_BadRefFlowId	int <b>Value:-62</b>
Constants.FE_FlowStructured	int <b>Value:-63</b>
Constants.FE_BadRefElementId	int <b>Value:-64</b>
Constants.FE_BadInsertPos	int <b>Value:-65</b>
Constants.FE_BadBookId	int <b>Value:-66</b>
Constants.FE_BookUnStructured	int <b>Value:-67</b>
Constants.FE_BadCompPath	int <b>Value:-68</b>
Constants.FE_BadElementSelection	int <b>Value:-69</b>
Constants.FE_FileClosedByClient	int <b>Value:-70</b>
Constants.FE_NotPgfOrFlow	int <b>Value:-71</b>
Constants.FE_NotMenu	int <b>Value:-72</b>
Constants.FE_NotCommand	int <b>Value:-73</b>
Constants.FE_NotApiCommand	int <b>Value:-74</b>



Property name	Data Type
Constants.FE_NotInMenu	int <b>Value:-75</b>
Constants.FE_BadShortcut	int <b>Value:-76</b>
Constants.FE_BadMenuBar	int <b>Value:-77</b>
Constants.FE_PropNotSet	int <b>Value:-78</b>
Constants.FE_InvAttributeDef	int <b>Value:-79</b>
Constants.FE_InvAttribute	int <b>Value:-80</b>
Constants.FE_CircularReference	int <b>Value:-81</b>
Constants.FE_NoSuchFlow	int <b>Value:-82</b>
Constants.FE_BadFileType	int <b>Value:-83</b>
Constants.FE_MissingFile	int <b>Value:-84</b>
Constants.FE_CantUpdateMacEdition	int <b>Value:-85</b>
Constants.FE_CanceledByClient	int <b>Value:-86</b>
Constants.FE_EmptyTextObject	int <b>Value:-87</b>
Constants.FE_Busy	int <b>Value:-88</b>
Constants.FE_FilterFailed	int <b>Value:-89</b>
Constants.FE_AsianSystemRequired	int <b>Value:-90</b>
Constants.FE_TintedColor	int <b>Value:-91</b>

Property name	Data Type
Constants.FE_NoColorFamily	int <b>Value:-92</b>
Constants.FE_StringTooLong	int <b>Value:-93</b>
Constants.FE_InternalErrorFailedToWrite Insets	int <b>Value:-94</b>
Constants.FE_NotFound	int <b>Value:-95</b>
Constants.FE_LanguageNotAvailable	int <b>Value:-96</b>
Constants.FE_BadBaseColor	int <b>Value:-97</b>
Constants.FE_BadFamilyName	int <b>Value:-98</b>
Constants.FE_BadInkName	int <b>Value:-99</b>
Constants.FE_ReservedColor	int <b>Value:-100</b>
Constants.FE_TableInLockedTi	int <b>Value:-101</b>
Constants.FE_XRefUnresolved	int <b>Value:-102</b>
Constants.FE_BadXRefSrcDocId	int <b>Value:-103</b>
Constants.FE_InvalidAttrExpr	int <b>Value:-107</b>
Constants.FE_DocAlreadyHasTrackedEdits	int <b>Value:-108</b>
Constants.FE_BookComponentAlreadyHasTrackedEdits	int <b>Value:-109</b>
Constants.FE_AcceptRejectCalledOnWrongPreviewState	int <b>Value:-110</b>
Constants.FE_FailedExportedFileInvalid	int <b>Value:-111</b>

Property name	Data Type
Constants.FE_BadKey	int <b>Value:-177</b>
Constants.FE_BadKeyField	int <b>Value:-178</b>
Constants.FE_BadValue	int <b>Value:-179</b>
Constants.FE_KeyCatalogNotLoaded	int <b>Value:-180</b>
Constants.FE_KeyCatalogIsStale	int <b>Value:-181</b>
Constants.FE_KeyDefinitionDoesNotExist	int <b>Value:-182</b>
Constants.FE_KeyDefinitionAlreadyExists	int <b>Value:-183</b>
Constants.FE_BadFilterType	int <b>Value:-184</b>
Constants.FE_KeyCatalogNotSpecified	int <b>Value:-185</b>
Constants.FE_NonDITADocument	int <b>Value:-186</b>
Constants.FE_UpdateDITARefereceFailedInvalidElementType	int <b>Value:-187</b>
Constants.FE_UpdateDITARefereceFailedCannotResolveReference	int <b>Value:-188</b>
Constants.FE_UpdateDITARefereceFailedCannotFindReferencedFile	int <b>Value:-189</b>
Constants.FE_UpdateDITARefereceFailedCannotOpenReferencedFile	int <b>Value:-190</b>
Constants.FE_UpdateDITARefereceFailedCannotConvertToFMOject	int <b>Value:-191</b>
Constants.FE_UpdateDITARefereceFailed	int <b>Value:-192</b>
Constants.FE_FDKUndoNotAllowed	int <b>Value:-193</b>

Property name	Data Type
Constants.FR_DialogStayUp	int <b>Value:-10000</b>
Constants.FR_ModalStayUp	int <b>Value:-10000</b>
Constants.FR_CancelOperation	int <b>Value:-10001</b>
Constants.FR_DisplayedTiDialog	int <b>Value:-10002</b>
Constants.FR_ECMImportSuccess	int <b>Value:-10003</b>
Constants.FR_SkipStep	int <b>Value:-10004</b>
Constants.FR_YesOperation	int <b>Value:-10005</b>
Constants.FR_NoOperation	int <b>Value:-10006</b>
Constants.FR_DisplayedXRefDialog	int <b>Value:-10007</b>
Constants.FR_ClosedXRefDialog	int <b>Value:-10008</b>
Constants.FR_DisplayedModelessDialogForNonContainerElem	int <b>Value:-10009</b>
Constants.FR_CancelInsertElementOperation	int <b>Value:-10010</b>
Constants.FR_HideDialogOnClose	int <b>Value:-10011</b>
Constants.FR_DisplayedModalDialogForNonContainerElem	int <b>Value:-10012</b>
Constants.FR_UpdateDITARreferenceSuccess	int <b>Value:-10013</b>
Constants.FR_UpdateDITARreferenceFailed	int <b>Value:-10014</b>
Constants.FR_UpdateDITARreferenceFailedInvalidElementType	int <b>Value:-10015</b>

Property name	Data Type
Constants.FR_UpdateDITAResourceFailedCannotResolveReference	int <b>Value:-10016</b>
Constants.FR_UpdateDITAResourceFailedCannotFindReferencedFile	int <b>Value:-10017</b>
Constants.FR_UpdateDITAResourceFailedCannotOpenReferencedFile	int <b>Value:-10018</b>
Constants.FR_UpdateDITAResourceFailedCannotConvertToFMObject	int <b>Value:-10019</b>
Constants.FR_KeyCatalogLoaded	int <b>Value:-10020</b>
Constants.FR_CommandEnabled	int <b>Value:-10021</b>
Constants.FR_CommandDisabled	int <b>Value:-10022</b>
Constants.FV_ACCESSBAR_OPEN	int <b>Value:1</b>
Constants.FV_ACCESSBAR_CLOSE	int <b>Value:2</b>
Constants.FV_MATH	int <b>Value:3</b>
Constants.FV_FORMATBAR_OPEN	int <b>Value:4</b>
Constants.FV_FORMATBAR_CLOSE	int <b>Value:5</b>
Constants.FV_CHAR_CATALOG	int <b>Value:6</b>
Constants.FV_PGF_CATALOG	int <b>Value:7</b>
Constants.FV_ELEMENT_CATALOG	int <b>Value:8</b>
Constants.FV_GRAPHIC	int <b>Value:9</b>
Constants.FV_THESAURUS	int <b>Value:10</b>

Property name	Data Type
Constants.FV_TEMPLATE_BROWSER	int <b>Value:11</b>
Constants.FV_HIST	int <b>Value:12</b>
Constants.FV_TBL_CATALOG	int <b>Value:13</b>
Constants.FV_GRAPHICS_CATALOG	int <b>Value:14</b>
Constants.FV_HELP_INDEX	int <b>Value:0</b>
Constants.FV_HELP_KEYS	int <b>Value:1</b>
Constants.FV_HELP_SAMPLES	int <b>Value:2</b>
Constants.FV_HELP_OVERVIEW	int <b>Value:3</b>
Constants.FV_HELP_ONLINE_MANUALS	int <b>Value:4</b>
Constants.FV_HELP_CONTEXT	int <b>Value:5</b>
Constants.FV_HELP_INIT_CONTEXT	int <b>Value:6</b>
Constants.FV_HELP_HYPertext	int <b>Value:7</b>
Constants.FV_HELP_WEBWORKS	int <b>Value:8</b>
Constants.FP_DockDialog	int <b>Value:2344</b>
Constants.FV_DIALOG_DOCK_NONE	int <b>Value:0x00</b>
Constants.FV_DIALOG_DOCK_LEFT	int <b>Value:0x1</b>
Constants.FV_DIALOG_DOCK_RIGHT	int <b>Value:(0x1 &lt;&lt; 1)</b>

Property name	Data Type
Constants.FV_DIALOG_DOCK_TOP	int <b>Value:</b> (0x1 << 2)
Constants.FV_DIALOG_DOCK_BOTTOM	int <b>Value:</b> (0x1 << 3)
Constants.FV_DIALOG_DOCK_ALL	int <b>Value:</b> 0xFF
Constants.FP_IsDialogDocked	int <b>Value:</b> 2345
Constants.FP_IsDialogVisible	int <b>Value:</b> 2346
Constants.FP_GroupDialog	int <b>Value:</b> 2349
Constants.FV_DIALOG_GROUP_NONE	int <b>Value:</b> 0x0
Constants.FV_DIALOG_GROUP_SPECIAL	int <b>Value:</b> 0x1
Constants.FV_DIALOG_GROUP_CATALOGS	int <b>Value:</b> (0x1 << 1)
Constants.FV_DIALOG_GROUP_DESIGNERS	int <b>Value:</b> (0x1 << 2)
Constants.FV_DIALOG_GROUP_ATTRIBUTES	int <b>Value:</b> (0x1 << 3)
Constants.FV_DIALOG_GROUP_PODS	int <b>Value:</b> (0x1 << 4)
Constants.FV_DIALOG_GROUP_PODSRIGHT	int <b>Value:</b> (0x1 << 5)
Constants.FV_DIALOG_GROUP_EDIT	int <b>Value:</b> (0x1 << 6)
Constants.FV_DIALOG_GROUP_ALLPANELS	int <b>Value:</b> (0x1 << 7)
Constants.FV_DIALOG_GROUP_RMKITS	int <b>Value:</b> (0x1 << 8)
Constants.FV_DIALOG_GROUP_RIGHTTOP	int <b>Value:</b> (0x1 << 9)

Property name	Data Type
Constants.FV_DIALOG_GROUP_RIGHT_BOTTOM	int <b>Value:</b> (0x1 << 10)
Constants.FV_DIALOG_GROUP_ALL	int <b>Value:</b> 0xFF
Constants.FP_ResizeGripperBarVisibility	int <b>Value:</b> 2395
Constants.FV_DIALOG_GRIPPERBAR_TRUE	int <b>Value:</b> 0
Constants.FV_DIALOG_GRIPPERBAR_FALSE	int <b>Value:</b> 1
Constants.FP_ResizeGripperPos	int <b>Value:</b> 2350
Constants.FV_DIALOG_RESIZEGRIPPERPOS_NONE	int <b>Value:</b> 0
Constants.FV_DIALOG_RESIZEGRIPPERPOS_FRONT	int <b>Value:</b> 1
Constants.FV_DIALOG_RESIZEGRIPPERPOS_BELOW	int <b>Value:</b> 2
Constants.FP_MinSize	int <b>Value:</b> 2351
Constants.FP_MaxSize	int <b>Value:</b> 2352
Constants.FP_Focus	int <b>Value:</b> 2358
Constants.FP_AlertClientUnique	int <b>Value:</b> 2290
Constants.FP_AlertClientName	int <b>Value:</b> 2291
Constants.FP_AlertString	int <b>Value:</b> 2292
Constants.FP_AlertType	int <b>Value:</b> 2293
Constants.FV_AlertOneButton	int <b>Value:</b> 1



Property name	Data Type
Constants.FV_AlertTwoButton	int <b>Value:2</b>
Constants.FV_AlertThreeButton	int <b>Value:3</b>
Constants.FF_ALERT_OK_DEFAULT	int <b>Value:0</b>
Constants.FF_ALERT_CANCEL_DEFAULT	int <b>Value:1</b>
Constants.FF_ALERT_CONTINUE_NOTE	int <b>Value:2</b>
Constants.FF_ALERT_CONTINUE_WARN	int <b>Value:3</b>
Constants.FF_ALERT_YES_DEFAULT	int <b>Value:4</b>
Constants.FF_ALERT_NO_DEFAULT	int <b>Value:5</b>
Constants.FF_ALERT_YES_NO_CANCEL	int <b>Value:6</b>
Constants.FF_XRUI_INTERNAL	int <b>Value:0x01</b>
Constants.FF_XRUI_OPEN_DOCS	int <b>Value:0x02</b>
Constants.FF_XRUI_CLOSED_DOCS	int <b>Value:0x04</b>
Constants.FF_XRUI_FORCE_UPDATE	int <b>Value:0x08</b>
Constants.FF_XRUI_EVERYTHING	int <b>Value:(0x01 0x02 0x04)</b>
Constants.FF_Custom_SingleSided	int <b>Value:0</b>
Constants.FF_Custom_FirstPageRight	int <b>Value:1</b>
Constants.FF_Custom_FirstPageLeft	int <b>Value:2</b>

Property name	Data Type
Constants.FF_CLOSE_MODIFIED	int <b>Value:1</b>
Constants.FF_CMP_SUMMARY_ONLY	int <b>Value:0x01</b>
Constants.FF_CMP_CHANGE_BARS	int <b>Value:0x02</b>
Constants.FF_CMP_HYPERLINKS	int <b>Value:0x04</b>
Constants.FF_CMP_SUMKIT	int <b>Value:0x08</b>
Constants.FF_CMP_COMPKIT	int <b>Value:0x10</b>
Constants.FF_CMP_ATTRIBUTES	int <b>Value:0x20</b>
Constants.FF_IMAGE_BACKGROUND	int <b>Value:0x0001</b>
Constants.FF_IMAGE_MASTER	int <b>Value:0x0002</b>
Constants.FF_IMAGE_CLIP	int <b>Value:0x0004</b>
Constants.FF_IMAGE_NOTRIM	int <b>Value:0x0008</b>
Constants.FF_IMAGE_PAGEFRAME	int <b>Value:0x000F</b>
Constants.FV_INTERACT_ROTATE	int <b>Value:1</b>
Constants.FV_INTERACT_ROTATE_CONSTRAIN	int <b>Value:2</b>
Constants.FF_ATTACH_BORDER_STATUS	int <b>Value:0x0001</b>
Constants.FF_ATTACH_BORDER_PAGENUM	int <b>Value:0x0002</b>
Constants.FF_ATTACH_BORDER_ZOOM	int <b>Value:0x0004</b>

Property name	Data Type
Constants.FF_ATTACH_BORDER_PAGEUP DOWN	int <b>Value:</b> 0x0008
Constants.FF_ATTACH_BORDER_RIGHTIC ONS	int <b>Value:</b> 0x0010
Constants.FF_ATTACH_ALL	int <b>Value:</b> (0x0001   0x0002   0x0004   0x0008   0x0010)
Constants.FF_UFF_PGF	int <b>Value:</b> 0x0001
Constants.FF_UFF_FONT	int <b>Value:</b> 0x0002
Constants.FF_UFF_PAGE	int <b>Value:</b> 0x0004
Constants.FF_UFF_TABLE	int <b>Value:</b> 0x0008
Constants.FF_UFF_COND	int <b>Value:</b> 0x0010
Constants.FF_UFF_REFPAGE	int <b>Value:</b> 0x0020
Constants.FF_UFF_VAR	int <b>Value:</b> 0x0040
Constants.FF_UFF_XREF	int <b>Value:</b> 0x0080
Constants.FF_UFF_COLOR	int <b>Value:</b> 0x0100
Constants.FF_UFF_MATH	int <b>Value:</b> 0x0200
Constants.FF_UFF_DOCUMENT_PROPS	int <b>Value:</b> 0x0400
Constants.FF_UFF_COMBINED_FONTS	int <b>Value:</b> 0x0800
Constants.FF_UFF_FBA	int <b>Value:</b> 0x1000
Constants.FF_UFF_REMOVE_PAGE_BREA KS	int <b>Value:</b> 0x4000

Property name	Data Type
Constants.FF_UFF_REMOVE_EXCEPTIONS	int Value:0x8000
Constants.FF_UFF_ALL	int Value:0xFFFF
Constants.FF_INTERACTIVE	int Value:0x0001
Constants.FF_CUT_TBL_CELLS	int Value:0x0002
Constants.FF_DONT_DELETE_HIDDEN_TEXT	int Value:0x0004
Constants.FF_INSERT_BELOW_RIGHT	int Value:0x0008
Constants.FF_VISIBLE_ONLY	int Value:0x0010
Constants.FF_REPLACE_CELLS	int Value:0x0020
Constants.FF_DONT_APPLY_ALL_ROWS	int Value:0x0040
Constants.FF_STRIP_HYPERTEXT	int Value:0x0080
Constants.FF_SELECT_WHOLE_TABLE	int Value:0x80000000
Constants.FF_CLEAR_SEL	int Value:0
Constants.FF_TEXT_SEL	int Value:1
Constants.FF_TEXT_EXT	int Value:2
Constants.FF_TEXT_Q_COPY	int Value:3
Constants.FF_CONTEXT_DRAGTXT	int Value:4
Constants.FF_PASTE_SEL_TEXT	int Value:11

Property name	Data Type
Constants.FF_PASTE_CLIP_TEXT	int <b>Value:12</b>
Constants.FF_COPY_TO_CLIP	int <b>Value:13</b>
Constants.FF_CELL_SEL	int <b>Value:21</b>
Constants.FF_CELL_EXT	int <b>Value:22</b>
Constants.FF_TABLE_SELALL	int <b>Value:23</b>
Constants.FF_CELL_RES	int <b>Value:24</b>
Constants.FF_CELLS_RES	int <b>Value:25</b>
Constants.FF_VIEWER_MENU	int <b>Value:32</b>
Constants.FF_CONTEXT_MENU	int <b>Value:33</b>
Constants.FF_STRUCTURE_MENU	int <b>Value:34</b>
Constants.FF_HYPertext	int <b>Value:41</b>
Constants.FF_ELEMENT_CONTEXT	int <b>Value:47</b>
Constants.FF_ELEMENT_EXT_CONTEXT	int <b>Value:48</b>
Constants.FF_BUBBLE_CONTEXT	int <b>Value:49</b>
Constants.FF_BUBBLE_EXT_CONTEXT	int <b>Value:50</b>
Constants.FF_ELEMENT_SEL	int <b>Value:51</b>
Constants.FF_ELEMENT_EXT	int <b>Value:52</b>

Property name	Data Type
Constants.FF_BUBBLE_SEL	int <b>Value:53</b>
Constants.FF_BUBBLE_EXT	int <b>Value:54</b>
Constants.FF_BUBBLE_COPY	int <b>Value:55</b>
Constants.FF_COLLAPSE_ELEM	int <b>Value:56</b>
Constants.FF_COLLAPSE_ALL	int <b>Value:57</b>
Constants.FF_OPEN_BOOK_COMP	int <b>Value:58</b>
Constants.FF_PASTE_CLIP_OBJ	int <b>Value:61</b>
Constants.FF_OBJ_SEL	int <b>Value:62</b>
Constants.FF_OBJ_EXT	int <b>Value:63</b>
Constants.FF_OBJ_EXT_BOR	int <b>Value:64</b>
Constants.FF_OBJ_Q_COPY	int <b>Value:65</b>
Constants.FF_OBJ_ROTATE	int <b>Value:66</b>
Constants.FF_OBJ_CROTATE	int <b>Value:67</b>
Constants.FF_BOR_SEL	int <b>Value:71</b>
Constants.FF_BOR_EXT	int <b>Value:72</b>
Constants.FF_CONTEXT_DRAG	int <b>Value:73</b>
Constants.FF_CONTEXT_DRAG_EXT	int <b>Value:74</b>

Property name	Data Type
Constants.FF_RES_MOVE	int <b>Value:81</b>
Constants.FF_RES_MOVE1	int <b>Value:82</b>
Constants.FF_RES_VERTEX	int <b>Value:83</b>
Constants.FF_RES_ADD	int <b>Value:84</b>
Constants.FF_RES_DEL	int <b>Value:85</b>
Constants.FF_SHIFT_KEY	int <b>Value:0x0001</b>
Constants.FF_CONTROL_KEY	int <b>Value:0x0002</b>
Constants.FF_ALT_KEY	int <b>Value:0x0004</b>
Constants.FF_CMD_KEY	int <b>Value:0x0010</b>
Constants.FF_ATTRDISP	int <b>Value:86</b>
Constants.FF_ATTRDISP_ALL	int <b>Value:87</b>
Constants.FF_EDIT_ATTRIBUTE	int <b>Value:88</b>
Constants.FF_ATTR_SEL	int <b>Value:89</b>
Constants.FS_ShowBrowser	int <b>Value:1</b>
Constants.FS_AlertUserAboutFailure	int <b>Value:2</b>
Constants.FS_DontNotifyAPIClients	int <b>Value:3</b>
Constants.FS_UpdateBrowserDirectory	int <b>Value:4</b>

Property name	Data Type
Constants.FS_MakeVisible	int <b>Value:5</b>
Constants.FS_MakeIconic	int <b>Value:6</b>
Constants.FS_DisallowMIF	int <b>Value:7</b>
Constants.FS_DisallowDoc	int <b>Value:8</b>
Constants.FS_DisallowBookMIF	int <b>Value:9</b>
Constants.FS_DisallowBookDoc	int <b>Value:10</b>
Constants.FS_DisallowFilterTypes	int <b>Value:11</b>
Constants.FS_DisallowPlainText	int <b>Value:12</b>
Constants.FS_ForceOpenAsText	int <b>Value:13</b>
Constants.FS_UseRecoverFile	int <b>Value:14</b>
Constants.FS_UseAutoSaveFile	int <b>Value:15</b>
Constants.FV_DoCancel	int <b>Value:0</b>
Constants.FV_DoOK	int <b>Value:1</b>
Constants.FV_DoYes	int <b>Value:2</b>
Constants.FV_DoNo	int <b>Value:3</b>
Constants.FV_DoShowDialog	int <b>Value:4</b>
Constants.FV_OpenViewOnly	int <b>Value:5</b>



Property name	Data Type
Constants.FS_FileIsText	int <b>Value:16</b>
Constants.FV_TextFile_EOLisEOP	int <b>Value:12</b>
Constants.FV_TextFile_EOLisNotEOP	int <b>Value:13</b>
Constants.FS_FileIsInUse	int <b>Value:17</b>
Constants.FV_ResetLockAndContinue	int <b>Value:7</b>
Constants.FV_OpenEditableCopy	int <b>Value:8</b>
Constants.FS_BookIsInUse	int <b>Value:18</b>
Constants.FS_LockCantBeReset	int <b>Value:19</b>
Constants.FS_FileIsOldVersion	int <b>Value:20</b>
Constants.FS_FileIsStructured	int <b>Value:21</b>
Constants.FV_StripStructureAndOpen	int <b>Value:10</b>
Constants.FS_FontNotFoundInDoc	int <b>Value:22</b>
Constants.FS_FontChangedMetric	int <b>Value:23</b>
Constants.FS_RefFileNotFound	int <b>Value:24</b>
Constants.FV_AllowAllRefFilesUnFindable	int <b>Value:11</b>
Constants.FS_LanguageNotAvailable	int <b>Value:25</b>
Constants.FS_OpenInNewWindow	int <b>Value:26</b>

Property name	Data Type
Constants.FS_OpenId	int <b>Value:27</b>
Constants.FS_FontNotFoundInCatalog	int <b>Value:28</b>
Constants.FS_NewDoc	int <b>Value:29</b>
Constants.FS_OpenDocViewOnly	int <b>Value:30</b>
Constants.FS_NameStripe	int <b>Value:31</b>
Constants.FS_BeefyDoc	int <b>Value:32</b>
Constants.FS_DisallowSgml	int <b>Value:33</b>
Constants.FS_FileTypeHint	int <b>Value:34</b>
Constants.FS_UpdateTextReferences	int <b>Value:35</b>
Constants.FS_UpdateXRefs	int <b>Value:36</b>
Constants.FS_OpenDocFluid	int <b>Value:37</b>
Constants.FV_DoUserPreference	int <b>Value:12</b>
Constants.FS_StructuredOpenApplication	int <b>Value:38</b>
Constants.FS_SgmlBookFileName	int <b>Value:39</b>
Constants.FS_OpenFileNotWritable	int <b>Value:40</b>
Constants.FS_OpenAsType	int <b>Value:41</b>
Constants.FV_AUTORECOGNIZE	int <b>Value:0x00</b>

Property name	Data Type
Constants.FV_TYPE_BINARY	int <b>Value:0x01</b>
Constants.FV_TYPE_MIF	int <b>Value:0x02</b>
Constants.FV_TYPE_TEXT	int <b>Value:0x03</b>
Constants.FV_TYPE_SGML	int <b>Value:0x04</b>
Constants.FV_TYPE_XML	int <b>Value:0x05</b>
Constants.FV_TYPE_FILTER	int <b>Value:0xFF</b>
Constants.FS_OpenBookViewOnly	int <b>Value:42</b>
Constants.FS_DisallowXml	int <b>Value:43</b>
Constants.FS_NoStructuredErrorLog	int <b>Value:44</b>
Constants.FS_VerifyInsets	int <b>Value:45</b>
Constants.FS_IsErrorDoc	int <b>Value:46</b>
Constants.FS_TemplateShouldInsertRoot	int <b>Value:47</b>
Constants.FS_CMSId	int <b>Value:48</b>
Constants.FS_CMSSessionId	int <b>Value:49</b>
Constants.FS_NumOpenParams	int <b>Value:49</b>
Constants.FS_OpenedFileName	int <b>Value:1</b>
Constants.FS_OpenNativeError	int <b>Value:2</b>

Property name	Data Type
Constants.FS_OpenStatus	int <b>Value:3</b>
Constants.FS_NumOpenReturnParams	int <b>Value:3</b>
Constants.FV_NumOpenStatusFields	int <b>Value:5</b>
Constants.FV_LockWasReset	int <b>Value:0</b>
Constants.FV_LockNotReset	int <b>Value:1</b>
Constants.FV_LockCouldntBeReset	int <b>Value:2</b>
Constants.FV_FileWasInUse	int <b>Value:3</b>
Constants.FV_FileIsViewOnly	int <b>Value:4</b>
Constants.FV_LockWasInvalid	int <b>Value:5</b>
Constants.FV_FileIsNotWritable	int <b>Value:6</b>
Constants.FV_FileModDateChanged	int <b>Value:7</b>
Constants.FV_FileHasNewName	int <b>Value:32</b>
Constants.FV_RecoverFileUsed	int <b>Value:33</b>
Constants.FV_AutoSaveFileUsed	int <b>Value:34</b>
Constants.FV_FileWasFiltered	int <b>Value:35</b>
Constants.FV_FontsWereMapped	int <b>Value:36</b>
Constants.FV_FontMetricsChanged	int <b>Value:37</b>

Property name	Data Type
Constants.FV_FontsMappedInCatalog	int <b>Value:38</b>
Constants.FV_LanguagesWerentFound	int <b>Value:39</b>
Constants.FV_BeefyDoc	int <b>Value:40</b>
Constants.FV_FileIsOldVersion	int <b>Value:41</b>
Constants.FV_FileStructureStripped	int <b>Value:42</b>
Constants.FV_FileIsText	int <b>Value:43</b>
Constants.FV_OpenedViewOnly	int <b>Value:44</b>
Constants.FV_EditableCopyOpened	int <b>Value:45</b>
Constants.FV_BadFileRefsWereMapped	int <b>Value:46</b>
Constants.FV_ReferencedFilesWerentFound	int <b>Value:47</b>
Constants.FV_FileAlreadyOpen	int <b>Value:48</b>
Constants.FV_UnresolvedXRefs	int <b>Value:49</b>
Constants.FV_UnresolvedTextInsets	int <b>Value:50</b>
Constants.FV_OpenedFluid	int <b>Value:51</b>
Constants.FV_FontsWithUnavailableEncodingsUsed	int <b>Value:53</b>
Constants.FV_FileHadStructure	int <b>Value:64</b>
Constants.FV_FileAlreadyOpenThisSession	int <b>Value:65</b>

Property name	Data Type
Constants.FV_BadFileType	int <b>Value:66</b>
Constants.FV_BadFileName	int <b>Value:67</b>
Constants.FV_CantNewBooks	int <b>Value:68</b>
Constants.FV_CantOpenBooksViewOnly	int <b>Value:69</b>
Constants.FV_BadScriptValue	int <b>Value:70</b>
Constants.FV_MissingScript	int <b>Value:71</b>
Constants.FV_CantForceOpenAsText	int <b>Value:72</b>
Constants.FV_DisallowedType	int <b>Value:73</b>
Constants.FV_DocDamagedByTextFilter	int <b>Value:74</b>
Constants.FV_DocHeadersDamaged	int <b>Value:75</b>
Constants.FV_DocWrongSize	int <b>Value:76</b>
Constants.FV_ChecksumDamage	int <b>Value:77</b>
Constants.FV_CantOpenBooksFluid	int <b>Value:78</b>
Constants.FV_CancelUseRecoverFile	int <b>Value:96</b>
Constants.FV_CancelUseAutoSaveFile	int <b>Value:97</b>
Constants.FV_CancelFileIsText	int <b>Value:98</b>
Constants.FV_CancelFileIsInUse	int <b>Value:99</b>

Property name	Data Type
Constants.FV_CancelFileHasStructure	int <b>Value:100</b>
Constants.FV_CancelReferencedFilesNot Found	int <b>Value:101</b>
Constants.FV_CancelLanguagesNot Found	int <b>Value:102</b>
Constants.FV_CancelFontsMapped	int <b>Value:103</b>
Constants.FV_CancelFontMetricsChanged	int <b>Value:104</b>
Constants.FV_CancelFontsMappedInCatalog	int <b>Value:105</b>
Constants.FV_CancelFileIsDoc	int <b>Value:106</b>
Constants.FV_CancelFileIsMIF	int <b>Value:107</b>
Constants.FV_CancelBook	int <b>Value:108</b>
Constants.FV_CancelBookMIF	int <b>Value:109</b>
Constants.FV_CancelFileIsFilterable	int <b>Value:110</b>
Constants.FV_CancelFileIsOldVersion	int <b>Value:111</b>
Constants.FV_UserCanceled	int <b>Value:112</b>
Constants.FV_CancelFileBrowser	int <b>Value:113</b>
Constants.FV_CancelBeefyDoc	int <b>Value:114</b>
Constants.FV_CancelFileIsSgml	int <b>Value:115</b>
Constants.FV_CancelFontsWithUnavailableEncodings	int <b>Value:116</b>

Property name	Data Type
Constants.FV_CancelOpenFileNotWritable	int <b>Value:117</b>
Constants.FV_CancelTempDiskFull	int <b>Value:118</b>
Constants.FV_CancelFileIsXml	int <b>Value:119</b>
Constants.FV_TooManyWindows	int <b>Value:128</b>
Constants.FV_BadTemplate	int <b>Value:129</b>
Constants.FV_FileNotReadable	int <b>Value:130</b>
Constants.FS_FileType	int <b>Value:1</b>
Constants.FV_SaveFmtBinary	int <b>Value:0</b>
Constants.FV_SaveFmtInterchange	int <b>Value:1</b>
Constants.FV_SaveFmtStationery	int <b>Value:3</b>
Constants.FV_SaveFmtViewOnly	int <b>Value:4</b>
Constants.FV_SaveFmtText	int <b>Value:6</b>
Constants.FV_SaveFmtSgml	int <b>Value:7</b>
Constants.FV_SaveFmtFilter	int <b>Value:8</b>
Constants.FV_SaveFmtPdf	int <b>Value:9</b>
Constants.FV_SaveFmtXml	int <b>Value:10</b>
Constants.FV_SaveFmtBinary60	int <b>Value:11</b>



Property name	Data Type
Constants.FV_SaveFmtBinary70	int <b>Value:12</b>
Constants.FV_SaveFmtBinary80	int <b>Value:13</b>
Constants.FV_SaveFmtInterchange70	int <b>Value:14</b>
Constants.FV_SaveFmtInterchange80	int <b>Value:15</b>
Constants.FV_SaveFmtBinary90	int <b>Value:16</b>
Constants.FV_SaveFmtInterchange90	int <b>Value:17</b>
Constants.FV_SaveFmtCompositeDoc	int <b>Value:18</b>
Constants.FV_SaveFmtBookWithXml	int <b>Value:19</b>
Constants.FV_SaveFmtBookWithFm	int <b>Value:20</b>
Constants.FV_SaveFmtBinary100	int <b>Value:21</b>
Constants.FV_SaveFmtInterchange100	int <b>Value:22</b>
Constants.FV_SaveFmtBinary110	int <b>Value:23</b>
Constants.FV_SaveFmtInterchange110	int <b>Value:24</b>
Constants.FV_SaveFmtBinary120	int <b>Value:25</b>
Constants.FV_SaveFmtInterchange120	int <b>Value:26</b>
Constants.FV_SaveFmtBinary130	int <b>Value:27</b>
Constants.FV_SaveFmtInterchange130	int <b>Value:28</b>

Property name	Data Type
Constants.FV_SaveFmtBinary140	int <b>Value:29</b>
Constants.FV_SaveFmtInterchange140	int <b>Value:30</b>
Constants.FV_SaveFmtBinary150	int <b>Value:31</b>
Constants.FV_SaveFmtInterchange150	int <b>Value:32</b>
Constants.FV_SaveFmtBinary160	int <b>Value:33</b>
Constants.FV_SaveFmtInterchange160	int <b>Value:34</b>
Constants.FV_SaveFmtBinary170	int <b>Value:35</b>
Constants.FV_SaveFmtInterchange170	int <b>Value:36</b>
Constants.FS_SaveMode	int <b>Value:4</b>
Constants.FV_ModeSave	int <b>Value:0</b>
Constants.FV_ModeSaveAs	int <b>Value:1</b>
Constants.FS_SaveAsModeName	int <b>Value:5</b>
Constants.FV_SaveAsNameAskUser	int <b>Value:0</b>
Constants.FV_SaveAsUseFileName	int <b>Value:1</b>
Constants.FV_SaveAsNameProvided	int <b>Value:2</b>
Constants.FS_AutoBackupOnSave	int <b>Value:6</b>
Constants.FV_SaveYesAutoBackup	int <b>Value:0</b>

Property name	Data Type
Constants.FV_SaveNoAutoBackup	int <b>Value:1</b>
Constants.FV_SaveUserPrefAutoBackup	int <b>Value:2</b>
Constants.FS_MakePageCount	int <b>Value:7</b>
Constants.FV_UseCurrentSetting	int <b>Value:0</b>
Constants.FV_DontChangePageCount	int <b>Value:1</b>
Constants.FV_MakePageCountEven	int <b>Value:2</b>
Constants.FV_MakePageCountOdd	int <b>Value:3</b>
Constants.FV_DeleteEmptyPages	int <b>Value:4</b>
Constants.FS_ShowSaveTextDialog	int <b>Value:8</b>
Constants.FS_SaveTextTblSetting	int <b>Value:9</b>
Constants.FV_SaveTblUserPref	int <b>Value:0</b>
Constants.FV_SaveTblRowsAsPgfs	int <b>Value:1</b>
Constants.FV_SaveTblColsAsPgfs	int <b>Value:2</b>
Constants.FV_SaveSkipTbIs	int <b>Value:3</b>
Constants.FS_SaveTextExtraBlankLineAtEOP	int <b>Value:10</b>
Constants.FS_UseDefaultUNIXpermissions	int <b>Value:11</b>
Constants.FS_RetainNameStripe	int <b>Value:12</b>

Property name	Data Type
Constants.FS_UNIXpermissions	int <b>Value:13</b>
Constants.FS_UpdateFRVList	int <b>Value:14</b>
Constants.FS_SaveFileTypeHint	int <b>Value:16</b>
Constants.FS_StructuredSaveApplication	int <b>Value:18</b>
Constants.FS_SaveFileNotWritable	int <b>Value:20</b>
Constants.FS_ModDateChanged	int <b>Value:21</b>
Constants.FS_DitavalFile	int <b>Value:22</b>
Constants.FS_DitavalCondTag	int <b>Value:23</b>
Constants.FS_DitaGenerateFlatBook	int <b>Value:25</b>
Constants.FS_DitaGenerateComponents AtOneLoc	int <b>Value:26</b>
Constants.FS_DitaPostProcessingOnBook	int <b>Value:27</b>
Constants.FS_DitaSavePdfViaBook	int <b>Value:28</b>
Constants.FS_HasUnsupportedHotspotS hapes	int <b>Value:29</b>
Constants.FS_NumSaveParams	int <b>Value:29</b>
Constants.FS_SavedFileName	int <b>Value:1</b>
Constants.FS_SaveNativeError	int <b>Value:2</b>
Constants.FS_SaveStatus	int <b>Value:3</b>

Property name	Data Type
Constants.FS_NumSaveReturnParams	int <b>Value:3</b>
Constants.FV_NumSaveStatusFields	int <b>Value:2</b>
Constants.FV_ProductIsViewer	int <b>Value:32</b>
Constants.FV_FileNotWritable	int <b>Value:33</b>
Constants.FV_BadSaveFileName	int <b>Value:34</b>
Constants.FV_BadFileId	int <b>Value:35</b>
Constants.FV_BadSaveScriptValue	int <b>Value:36</b>
Constants.FV_NonPortableSaveName	int <b>Value:37</b>
Constants.FV_NonPortableFileRefs	int <b>Value:38</b>
Constants.FV_ProductIsMaker	int <b>Value:39</b>
Constants.FV_BadSaveObjectId	int <b>Value:40</b>
Constants.FV_Unstructured	int <b>Value:41</b>
Constants.FV_InvalidSaveFilter	int <b>Value:42</b>
Constants.FV_UserCanceledSave	int <b>Value:48</b>
Constants.FV_FileWasExported	int <b>Value:49</b>
Constants.FV_CancelSaveFilesInUse	int <b>Value:50</b>
Constants.FV_CancelSaveFileNotWritable	int <b>Value:51</b>

Property name	Data Type
Constants.FV_CancelSaveModDateChanged	int <b>Value:52</b>
Constants.FV_CancelSaveExportedFileInvalid	int <b>Value:53</b>
Constants.FS_HowToImport	int <b>Value:4</b>
Constants.FV_DoByRef	int <b>Value:5</b>
Constants.FV_DoByCopy	int <b>Value:6</b>
Constants.FV_DoUserChoice	int <b>Value:7</b>
Constants.FS_ManualUpdate	int <b>Value:5</b>
Constants.FS_TextInsetName	int <b>Value:6</b>
Constants.FS_DisallowGraphicTypes	int <b>Value:9</b>
Constants.FS_DisallowMacEditions	int <b>Value:10</b>
Constants.FS_FileIsGraphic	int <b>Value:13</b>
Constants.FS_FitGraphicInSelectedRect	int <b>Value:14</b>
Constants.FS_GraphicDpi	int <b>Value:15</b>
Constants.FV_DoImportAsTable	int <b>Value:8</b>
Constants.FS_ForceImportAsText	int <b>Value:17</b>
Constants.FS_FileIsMakerDoc	int <b>Value:18</b>
Constants.FS_UseMainFlow	int <b>Value:19</b>

Property name	Data Type
Constants.FS_ImportFlowTag	int <b>Value:20</b>
Constants.FS_ImportFlowPageSpace	int <b>Value:21</b>
Constants.FV_ReferencePage	int <b>Value:9</b>
Constants.FV_BodyPage	int <b>Value:10</b>
Constants.FS_FormatImportedText	int <b>Value:22</b>
Constants.FV_SourceDoc	int <b>Value:0</b>
Constants.FV_EnclosingDoc	int <b>Value:1</b>
Constants.FV_PlainText	int <b>Value:2</b>
Constants.FS_RemoveManualPageBreaks	int <b>Value:23</b>
Constants.FS_RemoveOverrides	int <b>Value:24</b>
Constants.FS_ImportTblTag	int <b>Value:25</b>
Constants.FS_TblNumHeadingRows	int <b>Value:26</b>
Constants.FS_LeaveHeadingRowsEmpty	int <b>Value:27</b>
Constants.FS_TreatParaAsRow	int <b>Value:28</b>
Constants.FS_CellSeparator	int <b>Value:29</b>
Constants.FS_NumCellSeparators	int <b>Value:30</b>
Constants.FS_NumColumns	int <b>Value:31</b>

Property name	Data Type
Constants.FS_FileIsSgmlDoc	int <b>Value:32</b>
Constants.FS_StructuredImportApplication	int <b>Value:35</b>
Constants.FS_ImportAsType	int <b>Value:36</b>
Constants.FS_FileIsXmlDoc	int <b>Value:37</b>
Constants.FS_RasterDpi	int <b>Value:38</b>
Constants.FS_RasterImageWidth	int <b>Value:39</b>
Constants.FS_RasterImageHeight	int <b>Value:40</b>
Constants.FS_ShowRasterDpiDialog	int <b>Value:41</b>
Constants.FS_ClientUpdateSilent	int <b>Value:47</b>
Constants.FS_NumImportParams	int <b>Value:47</b>
Constants.FS_InsetData	int <b>Value:50</b>
Constants.FS_UseHTTP	int <b>Value:51</b>
Constants.FS_InsetElementDef	int <b>Value:52</b>
Constants.FS_PDFPageNum	int <b>Value:45</b>
Constants.FS_ImportedFileName	int <b>Value:1</b>
Constants.FS_ImportNativeError	int <b>Value:2</b>
Constants.FS_ImportStatus	int <b>Value:3</b>



Property name	Data Type
Constants.FS_NumImportReturnParams	int <b>Value:3</b>
Constants.FV_NumImportStatusFields	int <b>Value:2</b>
Constants.FV_ImportedByCopy	int <b>Value:0</b>
Constants.FV_ImportedText	int <b>Value:1</b>
Constants.FV_ImportedTextTable	int <b>Value:2</b>
Constants.FV_ImportedMIF	int <b>Value:3</b>
Constants.FV_ImportedMakerDoc	int <b>Value:4</b>
Constants.FV_ImportedFilteredFile	int <b>Value:5</b>
Constants.FV_ImportedGraphicFile	int <b>Value:6</b>
Constants.FV_ImportedMacEdition	int <b>Value:7</b>
Constants.FV_ImportedSgmlDoc	int <b>Value:8</b>
Constants.FV_ImportedXmlDoc	int <b>Value:9</b>
Constants.FV_BadImportFileName	int <b>Value:16</b>
Constants.FV_BadImportFileType	int <b>Value:17</b>
Constants.FV_BadImportScriptValue	int <b>Value:18</b>
Constants.FV_MissingImportScript	int <b>Value:19</b>
Constants.FV_CantForceImportAsText	int <b>Value:20</b>

Property name	Data Type
Constants.FV_DisallowedImportType	int <b>Value:21</b>
Constants.FV_NoMainFlow	int <b>Value:22</b>
Constants.FV_NoFlowWithSpecifiedName	int <b>Value:23</b>
Constants.FV_InsertionPointNotInText	int <b>Value:24</b>
Constants.FV_InsertionPointInTableCell	int <b>Value:25</b>
Constants.FV_InsertionPointInFootnote	int <b>Value:26</b>
Constants.FV_InsufficientMemory	int <b>Value:27</b>
Constants.FV_BadEnclosingDocId	int <b>Value:28</b>
Constants.FV_BadTextFileTypeHint	int <b>Value:29</b>
Constants.FV_FlowUnstructured	int <b>Value:30</b>
Constants.FV_CancelFileText	int <b>Value:32</b>
Constants.FV_CancelFileDoc	int <b>Value:33</b>
Constants.FV_CancelFileMIF	int <b>Value:34</b>
Constants.FV_CancelFileFilterable	int <b>Value:35</b>
Constants.FV_CancelFileGraphic	int <b>Value:36</b>
Constants.FV_UserCanceledImport	int <b>Value:37</b>
Constants.FV_CancelImportBrowser	int <b>Value:38</b>

Property name	Data Type
Constants.FV_CancelFileIsMacEdition	int <b>Value:39</b>
Constants.FV_CancelFileSgml	int <b>Value:40</b>
Constants.FV_CancelFileXml	int <b>Value:41</b>
Constants.FV_ImportFileNotReadable	int <b>Value:48</b>
Constants.FS_Export	int <b>Value:4</b>
Constants.FV_ExportDocument	int <b>Value:0</b>
Constants.FV_ExportObject	int <b>Value:1</b>
Constants.FS_ExportType	int <b>Value:5</b>
Constants.FS_ExportTextRange	int <b>Value:6</b>
Constants.FS_ExportObjectId	int <b>Value:7</b>
Constants.FS_ExportObjectNeedsDpi	int <b>Value:12</b>
Constants.FS_ExportFileTypeHint	int <b>Value:14</b>
Constants.FS_StructuredExportApplication	int <b>Value:16</b>
Constants.FS_ImportExportVersion	int <b>Value:17</b>
Constants.FS_NumExportParams	int <b>Value:17</b>
Constants.FS_ExportFileName	int <b>Value:1</b>
Constants.FS_ExportNativeError	int <b>Value:2</b>

Property name	Data Type
Constants.FS_ExportStatus	int <b>Value:3</b>
Constants.FV_NumExportStatusFields	int <b>Value:2</b>
Constants.FS_NumExportReturnParams	int <b>Value:3</b>
Constants.FV_ExportFileHasNewName	int <b>Value:0</b>
Constants.FV_ExportProductIsViewer	int <b>Value:1</b>
Constants.FV_ExportFileNotWritable	int <b>Value:2</b>
Constants.FV_InsufficientMem	int <b>Value:3</b>
Constants.FV_BadExportFileName	int <b>Value:16</b>
Constants.FV_MissingExportScript	int <b>Value:17</b>
Constants.FV_BadExportObjectId	int <b>Value:18</b>
Constants.FV_InvalidTextRange	int <b>Value:19</b>
Constants.FV_BadExportDocId	int <b>Value:20</b>
Constants.FV_BadExportScriptValue	int <b>Value:21</b>
Constants.FV_FilterFailed	int <b>Value:22</b>
Constants.FV_InvalidExportFilter	int <b>Value:23</b>
Constants.FV_UserCanceledExport	int <b>Value:32</b>
Constants.FV_CanceledExportObjectNeedsDpi	int <b>Value:33</b>

Property name	Data Type
Constants.FV_ApiClientCanceledExport	int <b>Value:34</b>
Constants.FS_AllowNonFMFiles	int <b>Value:1</b>
Constants.FS_AllowViewOnlyFiles	int <b>Value:3</b>
Constants.FS_ShowBookErrorLog	int <b>Value:4</b>
Constants.FS_AllowInconsistentNumPro ps	int <b>Value:6</b>
Constants.FS_UpdateBookGeneratedFile s	int <b>Value:7</b>
Constants.FS_UpdateBookNumbering	int <b>Value:8</b>
Constants.FS_UpdateBookOleLinks	int <b>Value:9</b>
Constants.FS_UpdateBookTextReference s	int <b>Value:10</b>
Constants.FS_UpdateBookXRefs	int <b>Value:11</b>
Constants.FS_UpdateBookMasterPages	int <b>Value:12</b>
Constants.FS_NumUpdateBookParams	int <b>Value:12</b>
Constants.FS_UpdateBookStatus	int <b>Value:1</b>
Constants.FS_NumUpdateBookReturnPa rams	int <b>Value:1</b>
Constants.FV_NumUpdateBookStatusFie lds	int <b>Value:4</b>
Constants.FV_BookNotSelfConsistent	int <b>Value:0</b>
Constants.FV_DuplicateFileInBook	int <b>Value:1</b>

Property name	Data Type
Constants.FV_NoNonGeneratedFilesInBook	int <b>Value:2</b>
Constants.FV_CancelNonFMFileInBook	int <b>Value:32</b>
Constants.FV_CancelViewOnlyFileInBook	int <b>Value:33</b>
Constants.FV_CancelInconsistentNumPr opsInFileInBook	int <b>Value:34</b>
Constants.FV_UserCanceledUpdateBook	int <b>Value:35</b>
Constants.FV_BadUpdateBookFileId	int <b>Value:64</b>
Constants.FV_BadUpdateBookScriptValu e	int <b>Value:65</b>
Constants.FV_FileInBookNotOpened	int <b>Value:96</b>
Constants.FV_FileInBookNotSaved	int <b>Value:97</b>
Constants.FV_TooManyWindowsUpdate Book	int <b>Value:98</b>
Constants.FV_Body	int <b>Value:1237</b>
Constants.FV_Heading	int <b>Value:1238</b>
Constants.FV_Footing	int <b>Value:1239</b>
Constants.FV_Above	int <b>Value:1240</b>
Constants.FV_Below	int <b>Value:1241</b>
Constants.FV_Left	int <b>Value:1242</b>
Constants.FV_Right	int <b>Value:1243</b>

Property name	Data Type
Constants.FV_ChooseSelect	int <b>Value:</b> 0x00000000
Constants.FV_ChooseOpen	int <b>Value:</b> 0x00000001
Constants.FV_ChooseSave	int <b>Value:</b> 0x00000002
Constants.FV_ChooseOpenDir	int <b>Value:</b> 0x00000004
Constants.FV_ChooseMultiSelect	int <b>Value:</b> 0x00000008
Constants.FV_ChooseMultiOpen	int <b>Value:</b> 0x00000010
Constants.FV_BrowseLocalOnly	int <b>Value:</b> 0x00010000
Constants.FV_BrowseCmsOnly	int <b>Value:</b> 0x00020000
Constants.FV_BrowseLocalWithWebdav Only	int <b>Value:</b> 0x00040000
Constants.FA_Init_First	int <b>Value:</b> 1
Constants.FA_Init_Subsequent	int <b>Value:</b> 2
Constants.FA_Init_TakeControl	int <b>Value:</b> 3
Constants.FA_Init_DocReport	int <b>Value:</b> 4
Constants.FA_Note_PreOpenDoc	int <b>Value:</b> 1
Constants.FA_Note_PostOpenDoc	int <b>Value:</b> 2
Constants.FA_Note_PreOpenMIF	int <b>Value:</b> 3
Constants.FA_Note_PostOpenMIF	int <b>Value:</b> 4

Property name	Data Type
Constants.FA_Note_PreSaveDoc	int <b>Value:5</b>
Constants.FA_Note_PostSaveDoc	int <b>Value:6</b>
Constants.FA_Note_PreSaveMIF	int <b>Value:7</b>
Constants.FA_Note_PostSaveMIF	int <b>Value:8</b>
Constants.FA_Note_PreFileType	int <b>Value:9</b>
Constants.FA_Note_PostFileType	int <b>Value:10</b>
Constants.FA_Note_PreQuitDoc	int <b>Value:11</b>
Constants.FA_Note_DirtyDoc	int <b>Value:12</b>
Constants.FA_Note_ClientCall	int <b>Value:13</b>
Constants.FA_Note_FilterIn	int <b>Value:14</b>
Constants.FA_Note_FilterOut	int <b>Value:15</b>
Constants.FA_Note_PreOpenBook	int <b>Value:16</b>
Constants.FA_Note_PostOpenBook	int <b>Value:17</b>
Constants.FA_Note_PreOpenBookMIF	int <b>Value:18</b>
Constants.FA_Note_PostOpenBookMIF	int <b>Value:19</b>
Constants.FA_Note_PreSaveBook	int <b>Value:20</b>
Constants.FA_Note_PostSaveBook	int <b>Value:21</b>



Property name	Data Type
Constants.FA_Note_PreSaveBookMIF	int <b>Value:22</b>
Constants.FA_Note_PostSaveBookMIF	int <b>Value:23</b>
Constants.FA_Note_PreQuitBook	int <b>Value:24</b>
Constants.FA_Note_DirtyBook	int <b>Value:25</b>
Constants.FA_Note_PreQuitSession	int <b>Value:26</b>
Constants.FA_Note_PostQuitSession	int <b>Value:27</b>
Constants.FA_Note_PreRevertDoc	int <b>Value:28</b>
Constants.FA_Note_PostRevertDoc	int <b>Value:29</b>
Constants.FA_Note_PreRevertBook	int <b>Value:30</b>
Constants.FA_Note_PostRevertBook	int <b>Value:31</b>
Constants.FA_Note_PreAutoSaveDoc	int <b>Value:32</b>
Constants.FA_Note_PostAutoSaveDoc	int <b>Value:33</b>
Constants.FA_Note_BackToUser	int <b>Value:34</b>
Constants.FA_Note_DisplayClientTiDialog	int <b>Value:35</b>
Constants.FA_Note_UpdateAllClientTi	int <b>Value:36</b>
Constants.FA_Note_UpdateClientTi	int <b>Value:37</b>
Constants.FA_Note_PreImport	int <b>Value:38</b>

Property name	Data Type
Constants.FA_Note_PostImport	int <b>Value:39</b>
Constants.FA_Note_PostQuitDoc	int <b>Value:40</b>
Constants.FA_Note_PostQuitBook	int <b>Value:41</b>
Constants.FA_Note_PreFunction	int <b>Value:42</b>
Constants.FA_Note_PostFunction	int <b>Value:43</b>
Constants.FA_Note_PreMouseCommand	int <b>Value:44</b>
Constants.FA_Note_PostMouseCommand	int <b>Value:45</b>
Constants.FA_Note_PreHypertext	int <b>Value:46</b>
Constants.FA_Note_PostHypertext	int <b>Value:47</b>
Constants.FA_Note_PrePrint	int <b>Value:48</b>
Constants.FA_Note_PostPrint	int <b>Value:49</b>
Constants.FA_Note_BodyPageAdded	int <b>Value:50</b>
Constants.FA_Note_BodyPageDeleted	int <b>Value:51</b>
Constants.FA_Note_PreInsertElement	int <b>Value:52</b>
Constants.FA_Note_PostInsertElement	int <b>Value:53</b>
Constants.FA_Note_PreChangeElement	int <b>Value:54</b>
Constants.FA_Note_PostChangeElement	int <b>Value:55</b>

Property name	Data Type
Constants.FA_Note_PreWrapElement	int <b>Value:56</b>
Constants.FA_Note_PostWrapElement	int <b>Value:57</b>
Constants.FA_Note_PreDragElement	int <b>Value:58</b>
Constants.FA_Note_PostDragElement	int <b>Value:59</b>
Constants.FA_Note_PreCopyElement	int <b>Value:60</b>
Constants.FA_Note_PostCopyElement	int <b>Value:61</b>
Constants.FA_Note_PreSetAttrValue	int <b>Value:62</b>
Constants.FA_Note_PostSetAttrValue	int <b>Value:63</b>
Constants.FA_Note_PreImportElemDefs	int <b>Value:64</b>
Constants.FA_Note_PostImportElemDefs	int <b>Value:65</b>
Constants.FA_Note_ECMInternal	int <b>Value:66</b>
Constants.FA_Note_PreExport	int <b>Value:67</b>
Constants.FA_Note_PostExport	int <b>Value:68</b>
Constants.FA_Note_PreInlineTypeIn	int <b>Value:69</b>
Constants.FA_Note_PostInlineTypeIn	int <b>Value:70</b>
Constants.FA_Note_PreSaveAsPDFDialog	int <b>Value:71</b>
Constants.FA_Note_PostSaveAsPDFDialog	int <b>Value:72</b>

Property name	Data Type
Constants.FA_Note_PreDistill	int <b>Value:73</b>
Constants.FA_Note_PostDistill	int <b>Value:74</b>
Constants.FA_Note_FilterFileToFile	int <b>Value:75</b>
Constants.FA_Note_PreBookComponent Open	int <b>Value:76</b>
Constants.FA_Note_PostBookComponent Open	int <b>Value:77</b>
Constants.FA_Note_PreGenerate	int <b>Value:78</b>
Constants.FA_Note_PostGenerate	int <b>Value:79</b>
Constants.FA_Note_PreGoToXrefSrc	int <b>Value:80</b>
Constants.FA_Note_PostGoToXrefSrc	int <b>Value:81</b>
Constants.FA_Note_PreOpenSGML	int <b>Value:82</b>
Constants.FA_Note_PostOpenSGML	int <b>Value:83</b>
Constants.FA_Note_Dialog	int <b>Value:84</b>
Constants.FA_Note_Alert	int <b>Value:85</b>
Constants.FA_Note_Palette	int <b>Value:86</b>
Constants.FA_Note_ToolBar	int <b>Value:87</b>
Constants.FA_Note_ConsoleMessage	int <b>Value:88</b>
Constants.FA_Note_Help	int <b>Value:89</b>

Property name	Data Type
Constants.FA_Note_URL	int <b>Value:90</b>
Constants.FA_Note_CursorChange	int <b>Value:91</b>
Constants.FA_Note_FontSubstitution	int <b>Value:92</b>
Constants.FA_Note_UndoCheckpoint	int <b>Value:93</b>
Constants.FA_Note_FileOpen	int <b>Value:94</b>
Constants.FA_Note_PreOpenXML	int <b>Value:95</b>
Constants.FA_Note_PostOpenXML	int <b>Value:96</b>
Constants.FA_Note_PreSaveXML	int <b>Value:97</b>
Constants.FA_Note_PostSaveXML	int <b>Value:98</b>
Constants.FA_Note_PreSaveSGML	int <b>Value:99</b>
Constants.FA_Note_PostSaveSGML	int <b>Value:100</b>
Constants.FA_Note_U3DCommand	int <b>Value:101</b>
Constants.FA_Note_Not_U3DCommand	int <b>Value:102</b>
Constants.FA_Note_Not_RSC_Supported_File	int <b>Value:103</b>
Constants.FA_Note_RSC_Supported_File	int <b>Value:104</b>
Constants.FA_Note_PostActiveDocChange	int <b>Value:105</b>
Constants.FA_Note_PreUpdateXRefs	int <b>Value:106</b>

Property name	Data Type
Constants.FA_Note_PostUpdateXRefs	int <b>Value:107</b>
Constants.FA_Note_DisplayClientXRefDialog	int <b>Value:108</b>
Constants.FA_Note_QuitModelessDialog	int <b>Value:109</b>
Constants.FA_Note_InsertTopicRef	int <b>Value:110</b>
Constants.FA_Note_InsertConRef	int <b>Value:111</b>
Constants.FA_Note_GenerateFM	int <b>Value:112</b>
Constants.FA_Note_OpenAllTopicrefs	int <b>Value:113</b>
Constants.FA_Note_UpdateRefs	int <b>Value:114</b>
Constants.FA_Note_AssignId	int <b>Value:115</b>
Constants.FA_Note_DITAOptions	int <b>Value:116</b>
Constants.FA_Note_NewDitamapFile	int <b>Value:117</b>
Constants.FA_Note_NewBookmapFile	int <b>Value:118</b>
Constants.FA_Note_NewTopicFile	int <b>Value:119</b>
Constants.FA_Note_NewTaskFile	int <b>Value:120</b>
Constants.FA_Note_NewConceptFile	int <b>Value:121</b>
Constants.FA_Note_NewReferenceFile	int <b>Value:122</b>
Constants.FA_Note_NewGlossEntryFile	int <b>Value:123</b>

Property name	Data Type
Constants.FA_Note_SWF_File	int <b>Value:124</b>
Constants.FA_Note_Not_SWF_File	int <b>Value:125</b>
Constants.FA_Note_Enable_Disable_DIT A_Menu_Commands	int <b>Value:126</b>
Constants.FA_Note_Dialog_Create	int <b>Value:127</b>
Constants.FA_Note_SetAttrValue	int <b>Value:128</b>
Constants.FA_Note_UpdateRefs_On_Save	int <b>Value:129</b>
Constants.FA_Note_Read_Struct_App	int <b>Value:130</b>
Constants.FA_Note_Struct_Wizard	int <b>Value:131</b>
Constants.FA_Note_Open_Dtd	int <b>Value:132</b>
Constants.FA_Note_Open_Schema	int <b>Value:133</b>
Constants.FA_Note_Open_EDD_For_App	int <b>Value:134</b>
Constants.FA_Note_PostSetPropertyValue	int <b>Value:135</b>
Constants.FA_Note_Not_AI_Supported_ File	int <b>Value:136</b>
Constants.FA_Note_AI_Supported_File	int <b>Value:137</b>
Constants.FA_Note_Poster_Applied	int <b>Value:138</b>
Constants.FA_Note_PrePublishDitamap	int <b>Value:139</b>
Constants.FA_Note_PostPublishDitamap	int <b>Value:140</b>

Property name	Data Type
Constants.FA_Note_UpdateDITAreferenc e	int <b>Value:</b> 141
Constants.FA_Note_UpdateDITAreferenc es	int <b>Value:</b> 142
Constants.FA_Note_LoadKeyCatalog	int <b>Value:</b> 143
Constants.FA_Note_ReLoadKeyCatalog	int <b>Value:</b> 144
Constants.FA_Note_EnumeratedCompon ent	int <b>Value:</b> 145
Constants.FA_Note_PreSwitchView	int <b>Value:</b> 146
Constants.FA_Note_PostSwitchView	int <b>Value:</b> 147
Constants.FA_Note_IsCommandEnabled	int <b>Value:</b> 148
Constants.FA_Note_Num	int <b>Value:</b> 149
Constants.FTI_String	int <b>Value:</b> 0x00000001
Constants.FTI_LineBegin	int <b>Value:</b> 0x00000002
Constants.FTI_LineEnd	int <b>Value:</b> 0x00000004
Constants.FTI_PgfBegin	int <b>Value:</b> 0x00000008
Constants.FTI_PgfEnd	int <b>Value:</b> 0x00000010
Constants.FTI_FlowBegin	int <b>Value:</b> 0x00000020
Constants.FTI_FlowEnd	int <b>Value:</b> 0x00000040
Constants.FTI_PageBegin	int <b>Value:</b> 0x00000080



Property name	Data Type
Constants.FTI_PageEnd	int Value:0x00000100
Constants.FTI_SubColBegin	int Value:0x00000200
Constants.FTI_SubColEnd	int Value:0x00000400
Constants.FTI_FrameAnchor	int Value:0x00000800
Constants.FTI_FnAnchor	int Value:0x00001000
Constants.FTI_TblAnchor	int Value:0x00002000
Constants.FTI_MarkerAnchor	int Value:0x00004000
Constants.FTI_XRefBegin	int Value:0x00008000
Constants.FTI_XRefEnd	int Value:0x00010000
Constants.FTI_VarBegin	int Value:0x00020000
Constants.FTI_VarEnd	int Value:0x00040000
Constants.FTI_ElementBegin	int Value:0x00080000
Constants.FTI_ElementEnd	int Value:0x00100000
Constants.FTI_CharPropsChange	int Value:0x00200000
Constants.FTI_TextFrameBegin	int Value:0x00400000
Constants.FTI_TextFrameEnd	int Value:0x00800000
Constants.FTI_TextObjId	int Value:0x01000000

Property name	Data Type
Constants.FTI_TextInsetBegin	int Value:0x02000000
Constants.FTI_TextInsetEnd	int Value:0x04000000
Constants.FTI_ElemPrefixBegin	int Value:0x08000000
Constants.FTI_ElemPrefixEnd	int Value:0x10000000
Constants.FTI_ElemSuffixBegin	int Value:0x20000000
Constants.FTI_ElemSuffixEnd	int Value:0x40000000
Constants.FTI2_RubiTextBegin	int Value:0x80000001
Constants.FTI2_RubiTextEnd	int Value:0x80000002
Constants.FTI2_RubiCompositeBegin	int Value:0x80000004
Constants.FTI2_RubiCompositeEnd	int Value:0x80000008
Constants.FTI_HardLineEnd	int Value:1
Constants.FTI_HyphenLineEnd	int Value:2
Constants.FTF_FAMILY	int Value:0x80000000
Constants.FTF_VARIATION	int Value:0x40000000
Constants.FTF_WEIGHT	int Value:0x20000000
Constants.FTF_ANGLE	int Value:0x10000000
Constants.FTF_UNDERLINING	int Value:0x08000000

Property name	Data Type
Constants.FTF_STRIKETHROUGH	int Value:0x04000000
Constants.FTF_OVERLINE	int Value:0x02000000
Constants.FTF_CHANGEBAR	int Value:0x01000000
Constants.FTF_OUTLINE	int Value:0x00800000
Constants.FTF_SHADOW	int Value:0x00400000
Constants.FTF_PAIRKERN	int Value:0x00200000
Constants.FTF_SIZE	int Value:0x00100000
Constants.FTF_KERNX	int Value:0x00080000
Constants.FTF_KERNY	int Value:0x00040000
Constants.FTF_SPREAD	int Value:0x00020000
Constants.FTF_COLOR	int Value:0x00010000
Constants.FTF_CHARTAG	int Value:0x00008000
Constants.FTF_CAPITALIZATION	int Value:0x00004000
Constants.FTF_POSITION	int Value:0x00002000
Constants.FTF_STRETCH	int Value:0x00001000
Constants.FTF_LANGUAGE	int Value:0x00000800
Constants.FTF_TSUME	int Value:0x00000400

Property name	Data Type
Constants.FTF_ENCODING	int <b>Value:</b> 0x00000200
Constants.FTF_IIF	int <b>Value:</b> 0x00000002
Constants.FTF_CONDITIONTAG	int <b>Value:</b> 0x00000001
Constants.FTF_ALL	int <b>Value:</b> 0xFFFFFE03
Constants.FV_OBJ_END_OFFSET	int <b>Value:</b> 0x50000000
Constants.FT_Bad	int <b>Value:</b> 0
Constants.FT_Integer	int <b>Value:</b> 1
Constants.FT_Metric	int <b>Value:</b> 2
Constants.FT_String	int <b>Value:</b> 3
Constants.FT_Id	int <b>Value:</b> 4
Constants.FT_Metrics	int <b>Value:</b> 5
Constants.FT_Strings	int <b>Value:</b> 6
Constants.FT_Points	int <b>Value:</b> 7
Constants.FT_Tabs	int <b>Value:</b> 8
Constants.FT_TextLoc	int <b>Value:</b> 9
Constants.FT_TextRange	int <b>Value:</b> 10
Constants.oldFT_ElementFmts	int <b>Value:</b> 11

Property name	Data Type
Constants.FT_ElementCatalog	int <b>Value:12</b>
Constants.FT_Ints	int <b>Value:13</b>
Constants.FT_UBytes	int <b>Value:14</b>
Constants.FT_UInts	int <b>Value:15</b>
Constants.FT_FormatBranches	int <b>Value:16</b>
Constants.FT_AttributeDefs	int <b>Value:17</b>
Constants.FT_Attributes	int <b>Value:18</b>
Constants.FT_ElementRange	int <b>Value:19</b>
Constants.FT_Vals	int <b>Value:20</b>
Constants.FT_AttributesEx	int <b>Value:21</b>
Constants.FT_Num	int <b>Value:22</b>
Constants.FI_PLUGIN_PRODUCTNAME	int <b>Value:1</b>
Constants.FI_PLUGIN_NAME	int <b>Value:2</b>
Constants.FI_PLUGIN_TYPE	int <b>Value:3</b>
Constants.FI_PLUGIN_DESCRIPTION	int <b>Value:4</b>
Constants.FI_PLUGIN_PRODUCTS	int <b>Value:5</b>
Constants.FI_PLUGIN_FACET	int <b>Value:6</b>

Property name	Data Type
Constants.FI_PLUGIN_FORMATID	int <b>Value:7</b>
Constants.FI_PLUGIN_VENDOR	int <b>Value:8</b>
Constants.FI_PLUGIN_SUFFIX	int <b>Value:9</b>
Constants.FI_PLUGIN_INFORMAT	int <b>Value:10</b>
Constants.FI_PLUGIN_OUTFORMAT	int <b>Value:11</b>
Constants.FO_Session	int <b>Value:0</b>
Constants.FO_StringResource	int <b>Value:1</b>
Constants.FO_Book	int <b>Value:2</b>
Constants.FO_BookComponent	int <b>Value:3</b>
Constants.FO_Doc	int <b>Value:4</b>
Constants.FO_BodyPage	int <b>Value:5</b>
Constants.FO_MasterPage	int <b>Value:6</b>
Constants.FO_RefPage	int <b>Value:7</b>
Constants.FO_HiddenPage	int <b>Value:8</b>
Constants.FO_Last_Page	int <b>Value:8</b>
Constants.FO_UnanchoredFrame	int <b>Value:9</b>
Constants.FO_Group	int <b>Value:10</b>

Property name	Data Type
Constants.FO_Arc	int <b>Value:11</b>
Constants.FO_Rectangle	int <b>Value:12</b>
Constants.FO_Ellipse	int <b>Value:13</b>
Constants.FO_RoundRect	int <b>Value:14</b>
Constants.FO_Polyline	int <b>Value:15</b>
Constants.FO_Polygon	int <b>Value:16</b>
Constants.FO_Line	int <b>Value:17</b>
Constants.FO_TextLine	int <b>Value:18</b>
Constants.FO_TextFrame	int <b>Value:19</b>
Constants.FO_Inset	int <b>Value:20</b>
Constants.FO_Math	int <b>Value:21</b>
Constants.FO_DBGGroup	int <b>Value:22</b>
Constants.FO_Last_Graphic	int <b>Value:22</b>
Constants.FO_AFrame	int <b>Value:23</b>
Constants.FO_XLast_Graphic	int <b>Value:23</b>
Constants.FO_Marker	int <b>Value:24</b>
Constants.FO_Fn	int <b>Value:25</b>

Property name	Data Type
Constants.FO_XRef	int <b>Value:26</b>
Constants.FO_XRefFmt	int <b>Value:27</b>
Constants.FO_Var	int <b>Value:28</b>
Constants.FO_VarFmt	int <b>Value:29</b>
Constants.FO_Flow	int <b>Value:30</b>
Constants.FO_Pgf	int <b>Value:31</b>
Constants.FO_PgfFmt	int <b>Value:32</b>
Constants.FO_CharFmt	int <b>Value:33</b>
Constants.FO_CondFmt	int <b>Value:34</b>
Constants.FO_Color	int <b>Value:35</b>
Constants.FO_Tbl	int <b>Value:36</b>
Constants.FO_TblFmt	int <b>Value:37</b>
Constants.FO_Row	int <b>Value:38</b>
Constants.FO_Cell	int <b>Value:39</b>
Constants.FO_RulingFmt	int <b>Value:40</b>
Constants.FO_ElementDef	int <b>Value:41</b>
Constants.FO_Element	int <b>Value:42</b>



Property name	Data Type
Constants.FO_DialogResource	int <b>Value:43</b>
Constants.FO_DlgBox	int <b>Value:44</b>
Constants.FO_DlgButton	int <b>Value:45</b>
Constants.FO_DlgTriBox	int <b>Value:46</b>
Constants.FO_DlgCheckBox	int <b>Value:47</b>
Constants.FO_DlgPopUp	int <b>Value:48</b>
Constants.FO_DlgScrollBar	int <b>Value:49</b>
Constants.FO_DlgEditBox	int <b>Value:50</b>
Constants.FO_DlgRadioButton	int <b>Value:51</b>
Constants.FO_DlgLabel	int <b>Value:52</b>
Constants.FO_DlgImage	int <b>Value:53</b>
Constants.FO_DlgScrollBar	int <b>Value:54</b>
Constants.FO_Menu	int <b>Value:55</b>
Constants.FO_Command	int <b>Value:56</b>
Constants.FO_MenuItemSeparator	int <b>Value:57</b>
Constants.FO_FmtChangeList	int <b>Value:58</b>
Constants.FO_FmtRule	int <b>Value:59</b>

Property name	Data Type
Constants.FO_FmtRuleClause	int <b>Value:60</b>
Constants.FO_TiFlow	int <b>Value:61</b>
Constants.FO_TiText	int <b>Value:62</b>
Constants.FO_TiTextTable	int <b>Value:63</b>
Constants.FO_TiApiClient	int <b>Value:64</b>
Constants.FO_SubCol	int <b>Value:65</b>
Constants.FO_MarkerType	int <b>Value:66</b>
Constants.FO_CombinedFontDefn	int <b>Value:67</b>
Constants.FO_Rubi	int <b>Value:68</b>
Constants.FO_Alert	int <b>Value:69</b>
Constants.FO_CursorResource	int <b>Value:70</b>
Constants.FO_AttrCondExpr	int <b>Value:71</b>
Constants.FO_DlgListView	int <b>Value:72</b>
Constants.FO_GraphicsFmt	int <b>Value:73</b>
Constants.FO_CMSRegistration	int <b>Value:74</b>
Constants.FO_CMSSession	int <b>Value:75</b>
Constants.FO_CMSObject	int <b>Value:76</b>

Property name	Data Type
Constants.FO_KeyCatalog	int <b>Value:</b> 77
Constants.FO_Iterator	int <b>Value:</b> 78
Constants.FO_Num	int <b>Value:</b> 79
Constants.FO_First_Internal	int <b>Value:</b> 79
Constants.FO_Page	int <b>Value:</b> (79+0)
Constants.FO_Graphic	int <b>Value:</b> (79+1)
Constants.FO_Frame	int <b>Value:</b> (79+2)
Constants.FO_Cblock	int <b>Value:</b> (79+3)
Constants.FO_SubCond	int <b>Value:</b> (79+4)
Constants.FO_SubStyle	int <b>Value:</b> (79+5)
Constants.FO_SubDash	int <b>Value:</b> (79+6)
Constants.FO_SubCell	int <b>Value:</b> (79+7)
Constants.FO_SubTbl	int <b>Value:</b> (79+8)
Constants.FO_SubPblock	int <b>Value:</b> (79+9)
Constants.FO_SubTextDef	int <b>Value:</b> (79+10)
Constants.FO_MenuCell	int <b>Value:</b> (79+11)
Constants.FO_TextInset	int <b>Value:</b> (79+12)

Property name	Data Type
Constants.FO_DitaMap	int <b>Value:</b> (79+13)
Constants.FO_Last_Internal	int <b>Value:</b> (79+14)
Constants.FO_CMSTConnection	int <b>Value:</b> (79+15)
Constants.FO_CMSTObject2	int <b>Value:</b> (79+16)
Constants.FO_Bad	int <b>Value:</b> 255
Constants.FO_AnchoredFrame	int <b>Value:</b> 23
Constants.FO_UFrame	int <b>Value:</b> 9
Constants.FP_Name	int <b>Value:</b> 20
Constants.FP_Unique	int <b>Value:</b> 21
Constants.FP_UserString	int <b>Value:</b> 22
Constants.FP_ObjectAttributes	int <b>Value:</b> 23
Constants.FP_Label	int <b>Value:</b> 30
Constants.FP_IsIconified	int <b>Value:</b> 31
Constants.FP_IsInFront	int <b>Value:</b> 32
Constants.FP_ScreenX	int <b>Value:</b> 33
Constants.FP_ScreenY	int <b>Value:</b> 34
Constants.FP_ScreenWidth	int <b>Value:</b> 35

Property name	Data Type
Constants.FP_ScreenHeight	int <b>Value:36</b>
Constants.FV_SessionId	int <b>Value:0</b>
Constants.FP_VersionRevision	int <b>Value:41</b>
Constants.FP_ProductName	int <b>Value:42</b>
Constants.FP_VersionMajor	int <b>Value:43</b>
Constants.FP_VersionMinor	int <b>Value:44</b>
Constants.FP_Platform	int <b>Value:45</b>
Constants.FP_OperatingSystem	int <b>Value:46</b>
Constants.FP_WindowSystem	int <b>Value:47</b>
Constants.FP_AutoSave	int <b>Value:48</b>
Constants.FP_AutoSaveSeconds	int <b>Value:49</b>
Constants.FP_AutoBackup	int <b>Value:50</b>
Constants.FP_ActiveDoc	int <b>Value:51</b>
Constants.FP_ProgId	int <b>Value:2421</b>
Constants.FP_IsFMRRunningAsServer	int <b>Value:2348</b>
Constants.FP_ActiveView	int <b>Value:2422</b>
Constants.FP_CurrentDoc	int <b>Value:2401</b>

Property name	Data Type
Constants.FP_PDFPageNum	int <b>Value:2402</b>
Constants.FP_UndoFDKRecording	int <b>Value:2403</b>
Constants.FP_StackWarningLevel	int <b>Value:2404</b>
Constants.FP_NoFlashInPDF	int <b>Value:2405</b>
Constants.FP_No3DInPDF	int <b>Value:2406</b>
Constants.FP_InsetURL	int <b>Value:2407</b>
Constants.FP_DontShowWelcomeScreen	int <b>Value:2408</b>
Constants.FP_TechSuiteInternal	int <b>Value:2409</b>
Constants.FP_IsTechnicalSuiteLicensed	int <b>Value:2410</b>
Constants.FP_TechSuiteInternal2	int <b>Value:2411</b>
Constants.FP_IsFMRunningInTrialPeriod	int <b>Value:2412</b>
Constants.FP_EnableAutoSpellCheck	int <b>Value:2413</b>
Constants.FP_CustomDelimiter	int <b>Value:2420</b>
Constants.FV_WarnNever	int <b>Value:0</b>
Constants.FV_WarnAlways	int <b>Value:1</b>
Constants.FV_WarnOnce	int <b>Value:2</b>
Constants.FV_UndoAll	int <b>Value:3</b>

Property name	Data Type
Constants.FV_RedoAll	int <b>Value:4</b>
Constants.FP_ActiveBook	int <b>Value:52</b>
Constants.FP_FirstOpenDoc	int <b>Value:53</b>
Constants.FP_FirstOpenBook	int <b>Value:54</b>
Constants.FP_FontFamilyNames	int <b>Value:55</b>
Constants.FP_FontVariationNames	int <b>Value:56</b>
Constants.FP_FontWeightNames	int <b>Value:57</b>
Constants.FP_FontAngleNames	int <b>Value:58</b>
Constants.FP_MarkerNames	int <b>Value:59</b>
Constants.FP_FontFamilyAttributes	int <b>Value:60</b>
Constants.FV_FAMILY_VISIBLE	int <b>Value:0x00000001</b>
Constants.FV_FAMILY_SELECTABLE	int <b>Value:0x00000002</b>
Constants.FV_FAMILY_MAPPED	int <b>Value:0x00000004</b>
Constants.FV_FAMILY_SURROGATE	int <b>Value:0xFFFF0000</b>
Constants.FP_RememberMissingFontNames	int <b>Value:61</b>
Constants.FP_Mif8bitOutput	int <b>Value:62</b>
Constants.FP_GreekSize	int <b>Value:63</b>

Property name	Data Type
Constants.FP_RetainUndoState	int <b>Value:64</b>
Constants.FP_ProductIsStructured	int <b>Value:65</b>
Constants.FP_ProductIsDemo	int <b>Value:66</b>
Constants.FP_DisableAutofitAfterZoom	int <b>Value:67</b>
Constants.FP_DefaultFontFamily	int <b>Value:68</b>
Constants.FP_DefaultFontVariation	int <b>Value:69</b>
Constants.FP_DefaultFontWeight	int <b>Value:70</b>
Constants.FP_DefaultFontAngle	int <b>Value:71</b>
Constants.FP_CTFontContext	int <b>Value:72</b>
Constants.FP_FontFamilyFullNames	int <b>Value:73</b>
Constants.FP_RpcPropertyName	int <b>Value:76</b>
Constants.FP_RpcProgramNumber	int <b>Value:77</b>
Constants.FP_DisplayName	int <b>Value:78</b>
Constants.FP_ProcessNumber	int <b>Value:79</b>
Constants.FP_OpenDir	int <b>Value:80</b>
Constants.FP_HostName	int <b>Value:81</b>
Constants.FP_UserName	int <b>Value:82</b>



Property name	Data Type
Constants.FP_UserLogin	int <b>Value:83</b>
Constants.FP_UserHomeDir	int <b>Value:84</b>
Constants.FP_UserSettingsDir	int <b>Value:152</b>
Constants.FP_Path	int <b>Value:85</b>
Constants.FP_TmpDir	int <b>Value:86</b>
Constants.FP_FM_HomeDir	int <b>Value:87</b>
Constants.FP_FM_BinDir	int <b>Value:88</b>
Constants.FP_FM_InitDir	int <b>Value:89</b>
Constants.FP_FM_CurrentDir	int <b>Value:90</b>
Constants.FP_FM_SgmlDir	int <b>Value:100</b>
Constants.FP_FM_StructureDir	int <b>Value:2031</b>
Constants.FP_FM_XmlDir	int <b>Value:2032</b>
Constants.FP_Reformatting	int <b>Value:91</b>
Constants.FP_Displaying	int <b>Value:92</b>
Constants.FP_ApplyFormatRules	int <b>Value:93</b>
Constants.FP_ApplyEOPRules	int <b>Value:2030</b>
Constants.FP_Validating	int <b>Value:94</b>

Property name	Data Type
Constants.FP_ToolBar	int <b>Value:95</b>
Constants.FP_FirstMenuItemInSession	int <b>Value:97</b>
Constants.FP_FirstCommandInSession	int <b>Value:98</b>
Constants.FP_CurrentMenuSet	int <b>Value:99</b>
Constants.FV_MENU_QUICK	int <b>Value:1</b>
Constants.FV_MENU_COMPLETE	int <b>Value:2</b>
Constants.FV_MENU_CUSTOM	int <b>Value:3</b>
Constants.FP_IconBarOn	int <b>Value:101</b>
Constants.FP_HelpPending	int <b>Value:102</b>
Constants.FP_FM_HelpDir	int <b>Value:103</b>
Constants.FP_PercentDone	int <b>Value:104</b>
Constants.FP_ActiveAlert	int <b>Value:105</b>
Constants.FP_Snap	int <b>Value:151</b>
Constants.FP_Gravity	int <b>Value:107</b>
Constants.FP_KByteAllocationSize	int <b>Value:106</b>
Constants.FP_ImportFilters	int <b>Value:2288</b>
Constants.FP_ExportFilters	int <b>Value:2289</b>

Property name	Data Type
Constants.FP_UIColorBackground	int <b>Value:2297</b>
Constants.FP_UIColorRuler	int <b>Value:2298</b>
Constants.FP_BIBGetAddressProc	int <b>Value:2299</b>
Constants.FP_RasterFilterDpi	int <b>Value:2400</b>
Constants.FV_Hint_HintVersion	int <b>Value:0</b>
Constants.FV_Hint_VendorId	int <b>Value:1</b>
Constants.FV_Hint_FormatId	int <b>Value:2</b>
Constants.FV_Hint_Platform	int <b>Value:3</b>
Constants.FV_Hint_FilterVersion	int <b>Value:4</b>
Constants.FV_Hint_FilterName	int <b>Value:5</b>
Constants.FP_DefaultKeyCatalog	int <b>Value:2871</b>
Constants.FP_FirstKeyCatalogInSession	int <b>Value:2872</b>
Constants.FP_KeyCatalogWorkflow	int <b>Value:2859</b>
Constants.FV_KeyCatalogWorkflowAuth oring	int <b>Value:0</b>
Constants.FV_KeyCatalogWorkflowPublis hing	int <b>Value:1</b>
Constants.FV_KeyCatalogWorkflowSearc hing	int <b>Value:2</b>
Constants.FV_KeyCatalogWorkflowMax	int <b>Value:3</b>

Property name	Data Type
Constants.FP_StatusLine	int <b>Value:108</b>
Constants.FP_TextSelection	int <b>Value:109</b>
Constants.FP_IsOnScreen	int <b>Value:110</b>
Constants.FP_NextOpenDocInSession	int <b>Value:111</b>
Constants.FP_FirstGraphicInDoc	int <b>Value:112</b>
Constants.FP_FirstPgflnDoc	int <b>Value:113</b>
Constants.FP_FirstMarkerInDoc	int <b>Value:114</b>
Constants.FP_FirstVarInDoc	int <b>Value:115</b>
Constants.FP_FirstVarFmtInDoc	int <b>Value:116</b>
Constants.FP_FirstXRefInDoc	int <b>Value:117</b>
Constants.FP_FirstXRefFmtInDoc	int <b>Value:118</b>
Constants.FP_FirstFnInDoc	int <b>Value:119</b>
Constants.FP_FirstTblInDoc	int <b>Value:120</b>
Constants.FP_FirstFlowInDoc	int <b>Value:121</b>
Constants.FP_FirstPgffmtInDoc	int <b>Value:122</b>
Constants.FP_FirstCharFmtInDoc	int <b>Value:123</b>
Constants.FP_FirstCondFmtInDoc	int <b>Value:124</b>

Property name	Data Type
Constants.FP_FirstTblFmtInDoc	int <b>Value:125</b>
Constants.FP_FirstRulingFmtInDoc	int <b>Value:126</b>
Constants.FP_FirstColorInDoc	int <b>Value:127</b>
Constants.FP_FirstSelectedGraphicInDoc	int <b>Value:128</b>
Constants.FP_SelectedTbl	int <b>Value:129</b>
Constants.FP_FileExtensionOverride	int <b>Value:2813</b>
Constants.FP_FirstAttrCondExprInDoc	int <b>Value:2815</b>
Constants.FP_DocOpenType	int <b>Value:130</b>
Constants.FV_DOC_TYPE_BINARY	int <b>Value:0x01</b>
Constants.FV_DOC_TYPE_TEXT	int <b>Value:0x02</b>
Constants.FV_DOC_TYPE_MIF	int <b>Value:0x03</b>
Constants.FV_DOC_TYPE_FILTER	int <b>Value:0x04</b>
Constants.FP_DocSaveFormat	int <b>Value:2415</b>
Constants.FP_DocIsModified	int <b>Value:131</b>
Constants.FP_DocIsHelp	int <b>Value:132</b>
Constants.FP_DocIsViewOnly	int <b>Value:133</b>
Constants.FP_ViewOnlyWinPalette	int <b>Value:134</b>

Property name	Data Type
Constants.FP_ViewOnlyWinMenubar	int Value:135
Constants.FP_ViewOnlyWinBorders	int Value:136
Constants.FP_ViewOnlyWinPopup	int Value:137
Constants.FP_ViewOnlyXRef	int Value:138
Constants.FV_VOX_NOT_ACTIVE	int Value:0
Constants.FV_VOX_GOTO_BEHAVIOR	int Value:1
Constants.FV_VOX_OPEN_BEHAVIOR	int Value:2
Constants.FV_VOX_ALERT	int Value:3
Constants.FP_ViewOnlySelect	int Value:139
Constants.FV_VOS_USER_ONLY	int Value:1
Constants.FV_VOS_NONE	int Value:2
Constants.FV_VOS_YES	int Value:3
Constants.FP_ViewOnlyDeadCodes	int Value:140
Constants.FP_FirstTiInDoc	int Value:141
Constants.FP_FirstSelectedTiInDoc	int Value:142
Constants.FP_DoclsViewOnlyWinPalette	int Value:134
Constants.FP_MenuBar	int Value:143

Property name	Data Type
Constants.FP_ViewOnlyMenuBar	int <b>Value:144</b>
Constants.FP_DocSaveType	int <b>Value:145</b>
Constants.FP_Untouchable	int <b>Value:146</b>
Constants.FP_DocFluidFlow	int <b>Value:147</b>
Constants.FP_FirstMarkerTypeInDoc	int <b>Value:148</b>
Constants.FP_MarkerTypeNames	int <b>Value:149</b>
Constants.FP_AddMarkerTypeToStandardMarkers	int <b>Value:150</b>
Constants.FP_ShowAll	int <b>Value:155</b>
Constants.FP_ShowCondIndicators	int <b>Value:156</b>
Constants.FP_DontUpdateXRefs	int <b>Value:159</b>
Constants.FP_DontUpdateTextInsets	int <b>Value:160</b>
Constants.FP_LineBreakAfter	int <b>Value:177</b>
Constants.FP_SuperScriptSize	int <b>Value:178</b>
Constants.FP_SuperScriptShift	int <b>Value:179</b>
Constants.FP_SubScriptSize	int <b>Value:180</b>
Constants.FP_SubScriptShift	int <b>Value:181</b>
Constants.FP_SmallCapsSize	int <b>Value:182</b>

Property name	Data Type
Constants.FP_SuperScriptStretch	int <b>Value:183</b>
Constants.FP_SubScriptStretch	int <b>Value:184</b>
Constants.FP_SmallCapsStretch	int <b>Value:185</b>
Constants.FP_RubiSize	int <b>Value:186</b>
Constants.FP_RubiFixedSize	int <b>Value:187</b>
Constants.FP_NarrowRubiSpaceForJapanese	int <b>Value:188</b>
Constants.FP_WideRubiSpaceForJapanese	int <b>Value:189</b>
Constants.FP_NarrowRubiSpaceForOther	int <b>Value:190</b>
Constants.FP_WideRubiSpaceForOther	int <b>Value:191</b>
Constants.FV_WIDE	int <b>Value:0</b>
Constants.FV_NARROW	int <b>Value:1</b>
Constants.FV_PROPORTIONAL	int <b>Value:2</b>
Constants.FP_RubiOverhang	int <b>Value:192</b>
Constants.FP_RubiAlignAtLineBounds	int <b>Value:193</b>
Constants.FP_FirstRubiInDoc	int <b>Value:194</b>
Constants.FP_ScreenCaptureDocToFile	int <b>Value:195</b>
Constants.FP_Dictionary	int <b>Value:203</b>



Property name	Data Type
Constants.FP_VolNumComputeMethod	int <b>Value:</b> 211
Constants.FV_NUM_READ_FROM_FILE	int <b>Value:</b> 0x00
Constants.FV_NUM_CONTINUE	int <b>Value:</b> 0x01
Constants.FV_NUM_RESTART	int <b>Value:</b> 0x02
Constants.FV_NUM_SAME	int <b>Value:</b> 0x03
Constants.FV_NUM_PERPAGE	int <b>Value:</b> 0x03
Constants.FP_VolumeNumber	int <b>Value:</b> 212
Constants.FP_VolumeNumStyle	int <b>Value:</b> 213
Constants.FP_VolumeNumText	int <b>Value:</b> 214
Constants.FP_ChapNumComputeMethod	int <b>Value:</b> 215
Constants.FP_ChapterNumber	int <b>Value:</b> 216
Constants.FP_ChapterNumStyle	int <b>Value:</b> 217
Constants.FV_NUMSTYLE_NUMERIC	int <b>Value:</b> 0x00
Constants.FV_NUMSTYLE_ROMAN_UC	int <b>Value:</b> 0x01
Constants.FV_NUMSTYLE_ROMAN_LC	int <b>Value:</b> 0x02
Constants.FV_NUMSTYLE_ALPHA_UC	int <b>Value:</b> 0x03
Constants.FV_NUMSTYLE_ALPHA_LC	int <b>Value:</b> 0x04

Property name	Data Type
Constants.FV_NUMSTYLE_KANJI	int Value:0x05
Constants.FV_NUMSTYLE_ZENKAKU	int Value:0x06
Constants.FV_NUMSTYLE_ZENKAKU_UC	int Value:0x07
Constants.FV_NUMSTYLE_ZENKAKU_LC	int Value:0x08
Constants.FV_NUMSTYLE_KANJI_KAZU	int Value:0x09
Constants.FV_NUMSTYLE_DAIJI	int Value:0x0a
Constants.FV_NUMSTYLE_TEXT	int Value:0x0b
Constants.FV_NUMSTYLE_FULLWIDTH	int Value:0x0c
Constants.FV_NUMSTYLE_FULLWIDTH_UC	int Value:0x0d
Constants.FV_NUMSTYLE_FULLWIDTH_LC	int Value:0x0e
Constants.FV_NUMSTYLE_CHINESE_NUMERIC	int Value:0x10
Constants.FP_ChapterNumText	int Value:218
Constants.FP_FirstPageNum	int Value:224
Constants.FP_PageNumStyle	int Value:225
Constants.FV_PAGE_NUM_NUMERIC	int Value:0x00
Constants.FV_PAGE_NUM_ROMAN_UC	int Value:0x01
Constants.FV_PAGE_NUM_ROMAN_LC	int Value:0x02

Property name	Data Type
Constants.FV_PAGE_NUM_ALPHA_UC	int Value:0x03
Constants.FV_PAGE_NUM_ALPHA_LC	int Value:0x04
Constants.FV_PAGE_NUM_KANJI	int Value:0x05
Constants.FV_PAGE_NUM_ZENKAKU	int Value:0x06
Constants.FV_PAGE_NUM_ZENKAKU_UC	int Value:0x07
Constants.FV_PAGE_NUM_ZENKAKU_LC	int Value:0x08
Constants.FV_PAGE_NUM_KANJI_KAZU	int Value:0x09
Constants.FV_PAGE_NUM_DAIJI	int Value:0x0a
Constants.FV_PAGE_NUM_FULLWIDTH	int Value:0x0c
Constants.FV_PAGE_NUM_FULLWIDTH_UC	int Value:0x0d
Constants.FV_PAGE_NUM_FULLWIDTH_LC	int Value:0x0e
Constants.FV_PAGE_NUM_CHINESE_NUMERIC	int Value:0x10
Constants.FP_DocIsDoubleSided	int Value:226
Constants.FP_FirstPageVerso	int Value:227
Constants.FP_PointPageNumStyle	int Value:228
Constants.FV_POINT_PAGE_NUM_NUMERIC	int Value:0x00
Constants.FV_POINT_PAGE_NUM_ROMAN_UC	int Value:0x01

Property name	Data Type
Constants.FV_POINT_PAGE_NUM_ROM AN_LC	int <b>Value:</b> 0x02
Constants.FV_POINT_PAGE_NUM_ALPH A_UC	int <b>Value:</b> 0x03
Constants.FV_POINT_PAGE_NUM_ALPH A_LC	int <b>Value:</b> 0x04
Constants.FV_POINT_PAGE_NUM_KANJI	int <b>Value:</b> 0x05
Constants.FV_POINT_PAGE_NUM_ZENK AKU	int <b>Value:</b> 0x06
Constants.FV_POINT_PAGE_NUM_ZENK AKU_UC	int <b>Value:</b> 0x07
Constants.FV_POINT_PAGE_NUM_ZENK AKU_LC	int <b>Value:</b> 0x08
Constants.FV_POINT_PAGE_NUM_KANJI _KAZU	int <b>Value:</b> 0x09
Constants.FV_POINT_PAGE_NUM_DAIJI	int <b>Value:</b> 0x0a
Constants.FV_POINT_PAGE_FULLWIDTH	int <b>Value:</b> 0x0c
Constants.FV_POINT_PAGE_FULLWIDTH _UC	int <b>Value:</b> 0x0d
Constants.FV_POINT_PAGE_FULLWIDTH _LC	int <b>Value:</b> 0x0e
Constants.FV_POINT_PAGE_CHINESE_N UMERIC	int <b>Value:</b> 0x10
Constants.FP_PageRounding	int <b>Value:</b> 229
Constants.FV_PR_DEL_EMPTY	int <b>Value:</b> 0x01
Constants.FV_PR_KEEP_NUM_EVEN	int <b>Value:</b> 0x02
Constants.FV_PR_KEEP_NUM_ODD	int <b>Value:</b> 0x03

Property name	Data Type
Constants.FV_PR_DONT_CHANGE	int <b>Value:</b> 0x04
Constants.FP_TopMargin	int <b>Value:</b> 230
Constants.FP_BottomMargin	int <b>Value:</b> 231
Constants.FP_LeftMargin	int <b>Value:</b> 232
Constants.FP_RightMargin	int <b>Value:</b> 233
Constants.FP_ColGap	int <b>Value:</b> 234
Constants.FP_NumCols	int <b>Value:</b> 235
Constants.FP_CurrentPage	int <b>Value:</b> 236
Constants.FP_SmartQuotes	int <b>Value:</b> 237
Constants.FP_AutoChangeBars	int <b>Value:</b> 238
Constants.FP_SmartSpaces	int <b>Value:</b> 239
Constants.FP_CurrentInset	int <b>Value:</b> 240
Constants.FP_ChangeBarDistance	int <b>Value:</b> 259
Constants.FP_ChangeBarPosition	int <b>Value:</b> 260
Constants.FV_CB_COL_LEFT	int <b>Value:</b> 0x00
Constants.FV_CB_COL_RIGHT	int <b>Value:</b> 0x01
Constants.FV_CB_COL_NEAREST	int <b>Value:</b> 0x02

Property name	Data Type
Constants.FV_CB_COL_FURTHEST	int <b>Value:</b> 0x03
Constants.FP_ChangeBarThickness	int <b>Value:</b> 261
Constants.FP_ChangeBarColor	int <b>Value:</b> 262
Constants.FP_FnFmt	int <b>Value:</b> 283
Constants.FP_FnFirstNum	int <b>Value:</b> 284
Constants.FP_FnCustNumString	int <b>Value:</b> 285
Constants.FP_FnRefPrefix	int <b>Value:</b> 286
Constants.FP_FnInstancePosition	int <b>Value:</b> 287
Constants.FV_FN_POS_SUPER	int <b>Value:</b> 0x00
Constants.FV_FN_POS_BASELINE	int <b>Value:</b> 0x01
Constants.FV_FN_POS_SUB	int <b>Value:</b> 0x02
Constants.FP_FnInstanceSuffix	int <b>Value:</b> 288
Constants.FP_FnHeightPerCol	int <b>Value:</b> 289
Constants.FP_FnNumStyle	int <b>Value:</b> 290
Constants.FV_FN_NUM_NUMERIC	int <b>Value:</b> 0x00
Constants.FV_FN_NUM_ROMAN_UC	int <b>Value:</b> 0x01
Constants.FV_FN_NUM_ROMAN_LC	int <b>Value:</b> 0x02

Property name	Data Type
Constants.FV_FN_NUM_ALPHA_UC	int <b>Value:</b> 0x03
Constants.FV_FN_NUM_ALPHA_LC	int <b>Value:</b> 0x04
Constants.FV_FN_NUM_CUSTOM	int <b>Value:</b> 0x05
Constants.FV_FN_NUM_KANJI	int <b>Value:</b> 0x06
Constants.FV_FN_NUM_ZENKAKU	int <b>Value:</b> 0x07
Constants.FV_FN_NUM_ZENKAKU_UC	int <b>Value:</b> 0x08
Constants.FV_FN_NUM_ZENKAKU_LC	int <b>Value:</b> 0x09
Constants.FV_FN_NUM_KANJI_KAZU	int <b>Value:</b> 0x0A
Constants.FV_FN_NUM_DAIJI	int <b>Value:</b> 0x0B
Constants.FV_FN_NUM_FULL_WIDTH	int <b>Value:</b> 0x0C
Constants.FV_FN_NUM_FULL_WIDTH_UC	int <b>Value:</b> 0x0D
Constants.FV_FN_NUM_FULL_WIDTH_LC	int <b>Value:</b> 0x0E
Constants.FV_FN_NUM_CHINESE_NUMERIC	int <b>Value:</b> 0x10
Constants.FP_FnNumberingPerPage	int <b>Value:</b> 291
Constants.FP_FnRefPosition	int <b>Value:</b> 292
Constants.FP_FnRefSuffix	int <b>Value:</b> 293
Constants.FP_FnInstancePrefix	int <b>Value:</b> 294

Property name	Data Type
Constants.FP_FnNumComputeMethod	int <b>Value:295</b>
Constants.FP_TblFnFmt	int <b>Value:315</b>
Constants.FP_TblFnNumStyle	int <b>Value:316</b>
Constants.FP_TblFnCustNumString	int <b>Value:317</b>
Constants.FP_TblFnCellPosition	int <b>Value:318</b>
Constants.FP_TblFnCellSuffix	int <b>Value:319</b>
Constants.FP_TblFnCellPrefix	int <b>Value:320</b>
Constants.FP_TblFnPosition	int <b>Value:321</b>
Constants.FP_TblFnPrefix	int <b>Value:322</b>
Constants.FP_TblFnSuffix	int <b>Value:323</b>
Constants.FP_TblFnNumComputeMethod	int <b>Value:324</b>
Constants.FP_Symbols	int <b>Value:334</b>
Constants.FP_SymbolsList	int <b>Value:335</b>
Constants.FP_Variables	int <b>Value:336</b>
Constants.FP_Strings	int <b>Value:337</b>
Constants.FP_Numbers	int <b>Value:338</b>
Constants.FP_Functions	int <b>Value:339</b>



Property name	Data Type
Constants.FP_HorizontalSpreadSmall	int <b>Value:340</b>
Constants.FP_HorizontalSpreadMed	int <b>Value:341</b>
Constants.FP_HorizontalSpreadLarge	int <b>Value:342</b>
Constants.FP_VerticalSpreadSmall	int <b>Value:343</b>
Constants.FP_VerticalSpreadMed	int <b>Value:344</b>
Constants.FP_VerticalSpreadLarge	int <b>Value:345</b>
Constants.FP_EqnIntegralSizeSmall	int <b>Value:346</b>
Constants.FP_EqnIntegralSizeMed	int <b>Value:347</b>
Constants.FP_EqnIntegralSizeLarge	int <b>Value:348</b>
Constants.FP_EqnSigmaSizeSmall	int <b>Value:349</b>
Constants.FP_EqnSigmaSizeMed	int <b>Value:350</b>
Constants.FP_EqnSigmaSizeLarge	int <b>Value:351</b>
Constants.FP_EqnLevel1SizeSmall	int <b>Value:352</b>
Constants.FP_EqnLevel1SizeMed	int <b>Value:353</b>
Constants.FP_EqnLevel1SizeLarge	int <b>Value:354</b>
Constants.FP_EqnLevel2SizeSmall	int <b>Value:355</b>
Constants.FP_EqnLevel2SizeMed	int <b>Value:356</b>

Property name	Data Type
Constants.FP_EqnLevel2SizeLarge	int <b>Value:357</b>
Constants.FP_EqnLevel3SizeSmall	int <b>Value:358</b>
Constants.FP_EqnLevel3SizeMed	int <b>Value:359</b>
Constants.FP_EqnLevel3SizeLarge	int <b>Value:360</b>
Constants.FP_ViewBorders	int <b>Value:382</b>
Constants.FP_ViewHotspotIndicators	int <b>Value:2357</b>
Constants.FP_ViewRulers	int <b>Value:383</b>
Constants.FP_ViewNoGraphics	int <b>Value:384</b>
Constants.FP_ViewPageScrolling	int <b>Value:385</b>
Constants.FV_SCROLL_VARIABLE	int <b>Value:0</b>
Constants.FV_SCROLL_HORIZONTAL	int <b>Value:1</b>
Constants.FV_SCROLL_VERTICAL	int <b>Value:2</b>
Constants.FV_SCROLL_FACING	int <b>Value:3</b>
Constants.FP_ViewGridUnits	int <b>Value:386</b>
Constants.FP_Zoom	int <b>Value:387</b>
Constants.FV_NO_ZOOM_MANGLING	int <b>Value:-1</b>
Constants.FV_ZOOM_TO_FIT_TO_WINDOW	int <b>Value:-2</b>

Property name	Data Type
Constants.FV_FIT_WINDOW_TO_PAGE	int <b>Value:-3</b>
Constants.FP_ViewTextSymbols	int <b>Value:388</b>
Constants.FP_ViewGrid	int <b>Value:389</b>
Constants.FP_ViewDisplayUnits	int <b>Value:390</b>
Constants.FP_TrackChangesOn	int <b>Value:2819</b>
Constants.FP_PreviewState	int <b>Value:2820</b>
Constants.FV_PREVIEW_OFF_TRACK_CHANGE	int <b>Value:0</b>
Constants.FV_PREVIEW_ON_ORIGINAL	int <b>Value:1</b>
Constants.FV_PREVIEW_ON_FINAL	int <b>Value:2</b>
Constants.FP_ReviewerNameList	int <b>Value:2825</b>
Constants.FP_TrackChangesAddedColor	int <b>Value:2826</b>
Constants.FP_TrackChangesDeletedColor	int <b>Value:2827</b>
Constants.FP_ViewRulerUnits	int <b>Value:391</b>
Constants.FP_SpotColorView	int <b>Value:392</b>
Constants.FP_SnapGridUnits	int <b>Value:393</b>
Constants.FP_SnapAngle	int <b>Value:394</b>
Constants.FP_ViewLinkBoundaries	int <b>Value:395</b>

Property name	Data Type
Constants.FP_ViewFontSizeUnits	int <b>Value:396</b>
Constants.FP_LeftMasterPage	int <b>Value:413</b>
Constants.FP_RightMasterPage	int <b>Value:414</b>
Constants.FP_FirstBodyPageInDoc	int <b>Value:415</b>
Constants.FP_LastBodyPageInDoc	int <b>Value:416</b>
Constants.FP_FirstMasterPageInDoc	int <b>Value:417</b>
Constants.FP_LastMasterPageInDoc	int <b>Value:418</b>
Constants.FP_FirstRefPageInDoc	int <b>Value:419</b>
Constants.FP_LastRefPageInDoc	int <b>Value:420</b>
Constants.FP_HiddenPage	int <b>Value:421</b>
Constants.FP_MainFlowInDoc	int <b>Value:422</b>
Constants.FP_PrintStartPageName	int <b>Value:439</b>
Constants.FP_PrintEndPageName	int <b>Value:440</b>
Constants.FP_PrintCols	int <b>Value:441</b>
Constants.FP_PrintRows	int <b>Value:442</b>
Constants.FP_PrintStartPage	int <b>Value:443</b>
Constants.FP_PrintStartPoint	int <b>Value:444</b>

Property name	Data Type
Constants.FP_PrintOddPages	int <b>Value:445</b>
Constants.FP_PrintCollated	int <b>Value:446</b>
Constants.FP_PrintLowRes	int <b>Value:447</b>
Constants.FP_PrintThumbnails	int <b>Value:448</b>
Constants.FP_PrinterName	int <b>Value:449</b>
Constants.FP_PrintScale	int <b>Value:450</b>
Constants.FP_PrintFileName	int <b>Value:451</b>
Constants.FP_PrintDitavalFileName	int <b>Value:2343</b>
Constants.FP_PrintScope	int <b>Value:452</b>
Constants.FV_PR_ALL	int <b>Value:1</b>
Constants.FV_PR_RANGE	int <b>Value:2</b>
Constants.FP_PrintEndPage	int <b>Value:453</b>
Constants.FP_PrintEndPoint	int <b>Value:454</b>
Constants.FP_PrintEvenPages	int <b>Value:455</b>
Constants.FP_PrintLastSheetFirst	int <b>Value:456</b>
Constants.FP_PrintRegistrationMarks	int <b>Value:457</b>
Constants.FP_PrintManualFeed	int <b>Value:458</b>

Property name	Data Type
Constants.FP_PrintNumCopies	int <b>Value:</b> 459
Constants.FP_PrintToFile	int <b>Value:</b> 460
Constants.FP_PrintPaperWidth	int <b>Value:</b> 461
Constants.FP_PrintPaperHeight	int <b>Value:</b> 462
Constants.FP_PrintSeps	int <b>Value:</b> 463
Constants.FP_SkipBlankSeps	int <b>Value:</b> 464
Constants.FP_PrintImaging	int <b>Value:</b> 465
Constants.FV_IMG_POSITIVE	int <b>Value:</b> 0
Constants.FV_IMG_NEGATIVE	int <b>Value:</b> 1
Constants.FP_PrintEmulsion	int <b>Value:</b> 466
Constants.FV_EMUL_UP	int <b>Value:</b> 0
Constants.FV_EMUL_DOWN	int <b>Value:</b> 1
Constants.FP_PrintBlankPages	int <b>Value:</b> 467
Constants.FP_PrintTomboMarks	int <b>Value:</b> 468
Constants.FP_PrintRegMarkDate	int <b>Value:</b> 469
Constants.FP_TrapwiseCompatibility	int <b>Value:</b> 495
Constants.FP_DownloadFonts	int <b>Value:</b> 496

Property name	Data Type
Constants.FP_PrintSpotBW	int <b>Value:498</b>
Constants.FV_PR_DOWNLOAD_NONE	int <b>Value:1</b>
Constants.FV_PR_DOWNLOAD_ALL	int <b>Value:2</b>
Constants.FV_PR_DOWNLOAD_ALL_BUT_STANDARD_13	int <b>Value:3</b>
Constants.FV_PR_DOWNLOAD_ALL_BUT_STANDARD_35	int <b>Value:4</b>
Constants.FP_PrintDownloadAsianFonts	int <b>Value:501</b>
Constants.FP_PrintDownloadTrueTypeAsType1	int <b>Value:502</b>
Constants.FP_GenerateAcrobatInfo	int <b>Value:2250</b>
Constants.FP_AcrobatBookmarkDisplayTags	int <b>Value:2251</b>
Constants.FP_DocAcrobatDefaultsChanged	int <b>Value:2252</b>
Constants.FP_DocAcrobatElements	int <b>Value:2253</b>
Constants.FP_DocAcrobatElementList	int <b>Value:2254</b>
Constants.FP_DocAcrobatNoArticleThreads	int <b>Value:2255</b>
Constants.FP_DocAcrobatColumnArticleThreads	int <b>Value:2256</b>
Constants.FP_PDFAllNamedDestinations	int <b>Value:2257</b>
Constants.FP_PDFDestsMarked	int <b>Value:2258</b>
Constants.FP_PDFStructure	int <b>Value:2259</b>

Property name	Data Type
Constants.FP_PDFDocInfo	int <b>Value:2260</b>
Constants.FP_PDFBookmark	int <b>Value:2261</b>
Constants.FP_FileInfoPacket	int <b>Value:2262</b>
Constants.FP_PDFJobOption	int <b>Value:2263</b>
Constants.FP_PDFOpenPage	int <b>Value:2264</b>
Constants.FP_PDFZoomType	int <b>Value:2265</b>
Constants.FV_PDFZoomNone	int <b>Value:0</b>
Constants.FV_PDFZoomDefault	int <b>Value:1</b>
Constants.FV_PDFZoomPage	int <b>Value:2</b>
Constants.FV_PDFZoomWidth	int <b>Value:3</b>
Constants.FV_PDFZoomHeight	int <b>Value:4</b>
Constants.FV_PDFZoomMaxValue	int <b>Value:4</b>
Constants.FV_PDFRegistrationMarksNone	int <b>Value:0</b>
Constants.FV_PDFRegistrationMarksWestern	int <b>Value:1</b>
Constants.FV_PDFRegistrationMarksTomb	int <b>Value:2</b>
Constants.FV_PDFRegistrationMarksMax	int <b>Value:2</b>
Constants.FP_PDFZoomFactor	int <b>Value:2266</b>



Property name	Data Type
Constants.FP_PDFSeparateFiles	int <b>Value:</b> 2267
Constants.FP_PDFRegistrationMarks	int <b>Value:</b> 2268
Constants.FP_PDFPageWidth	int <b>Value:</b> 2269
Constants.FP_PDFPageHeight	int <b>Value:</b> 2270
Constants.FP_PDFPrintPageRange	int <b>Value:</b> 2271
Constants.FP_PDFStartPage	int <b>Value:</b> 2272
Constants.FP_PDFEndPage	int <b>Value:</b> 2273
Constants.FP_PDFConvertCMYKtoRGB	int <b>Value:</b> 2274
Constants.FP_PDFBookmarksOpenLevel	int <b>Value:</b> 2275
Constants.FP_PDFDistillerAbsent	int <b>Value:</b> 2276
Constants.FP_PDFJobOptionsAbsent	int <b>Value:</b> 2277
Constants.FP_PDFViewPDF	int <b>Value:</b> 2278
Constants.FP_PDFGenerateForReview	int <b>Value:</b> 2279
Constants.FV_PDFBookmarksOpenDefaultLevel	int <b>Value:</b> -1
Constants.FV_PDFBookmarksOpenAllLevels	int <b>Value:</b> -2
Constants.FV_PDFBookmarksOpenNoneLevel	int <b>Value:</b> -3
Constants.FP_MaxFirstIndent	int <b>Value:</b> 472

Property name	Data Type
Constants.FP_MinFirstIndent	int <b>Value:473</b>
Constants.FP_MaxLeftIndent	int <b>Value:474</b>
Constants.FP_MinLeftIndent	int <b>Value:475</b>
Constants.FP_MaxRightIndent	int <b>Value:476</b>
Constants.FP_MinRightIndent	int <b>Value:477</b>
Constants.FP_MaxSpaceAbove	int <b>Value:478</b>
Constants.FP_MinSpaceAbove	int <b>Value:479</b>
Constants.FP_MaxSpaceBelow	int <b>Value:480</b>
Constants.FP_MinSpaceBelow	int <b>Value:481</b>
Constants.FP_MaxLeading	int <b>Value:482</b>
Constants.FP_MinLeading	int <b>Value:483</b>
Constants.FP_MaxFontSize	int <b>Value:484</b>
Constants.FP_MinFontSize	int <b>Value:485</b>
Constants.FP_MaxSpread	int <b>Value:486</b>
Constants.FP_MinSpread	int <b>Value:487</b>
Constants.FP_MaxTabPosition	int <b>Value:701</b>
Constants.FP_MinTabPosition	int <b>Value:702</b>

Property name	Data Type
Constants.FP_MaxLeftMargin	int <b>Value:703</b>
Constants.FP_MinLeftMargin	int <b>Value:704</b>
Constants.FP_MaxRightMargin	int <b>Value:705</b>
Constants.FP_MinRightMargin	int <b>Value:706</b>
Constants.FP_MaxTopMargin	int <b>Value:707</b>
Constants.FP_MinTopMargin	int <b>Value:708</b>
Constants.FP_MaxBottomMargin	int <b>Value:709</b>
Constants.FP_MinBottomMargin	int <b>Value:710</b>
Constants.FP_MaxStretch	int <b>Value:711</b>
Constants.FP_MinStretch	int <b>Value:712</b>
Constants.FP_MagicMarker	int <b>Value:488</b>
Constants.FP_XmlVersion	int <b>Value:2800</b>
Constants.FP_XmlEncoding	int <b>Value:2801</b>
Constants.FP_XmlStandAlone	int <b>Value:2802</b>
Constants.FV_XML_STANDALONE_YES	int <b>Value:1</b>
Constants.FV_XML_STANDALONE_NO	int <b>Value:2</b>
Constants.FV_XML_STANDALONE_NONE	int <b>Value:3</b>

Property name	Data Type
Constants.FV_XML_STANDALONE_NODE C	int <b>Value:4</b>
Constants.FP_XmlStyleSheet	int <b>Value:2803</b>
Constants.FP_XmlStyleSheetList	int <b>Value:2804</b>
Constants.FP_XmlUseBOM	int <b>Value:2805</b>
Constants.FV_XML_USEBOM_YES	int <b>Value:1</b>
Constants.FV_XML_USEBOM_NO	int <b>Value:2</b>
Constants.FV_XML_USEBOM_UTF8	int <b>Value:3</b>
Constants.FV_XML_USEBOM_UTF16BE	int <b>Value:4</b>
Constants.FV_XML_USEBOM_UTF16LE	int <b>Value:5</b>
Constants.FV_XML_USEBOM_UTF32BE	int <b>Value:6</b>
Constants.FV_XML_USEBOM_UTF32LE	int <b>Value:7</b>
Constants.FP_XmlWellFormed	int <b>Value:2806</b>
Constants.FV_XML_WELLFORMED_YES	int <b>Value:1</b>
Constants.FV_XML_WELLFORMED_NO	int <b>Value:2</b>
Constants.FP_XmlFileEncoding	int <b>Value:2809</b>
Constants.FP_XmlDocType	int <b>Value:2810</b>
Constants.FP_XmlPublicId	int <b>Value:2811</b>

Property name	Data Type
Constants.FP_XmlSystemId	int <b>Value:2812</b>
Constants.FP_XmlLock	int <b>Value:2841</b>
Constants.FP_ServerUrl	int <b>Value:2807</b>
Constants.FP_ServerState	int <b>Value:2808</b>
Constants.FV_URL_CHECKED_IN	int <b>Value:1</b>
Constants.FV_URL_CHECKED_OUT	int <b>Value:2</b>
Constants.FP_KeyCatalog	int <b>Value:2856</b>
Constants.FP_KeyCatalogType	int <b>Value:2857</b>
Constants.FP_SpecifiedKeyCatalog	int <b>Value:2858</b>
Constants.FP_NextOpenBookInSession	int <b>Value:490</b>
Constants.FP_FirstComponentInBook	int <b>Value:491</b>
Constants.FP_BooksModified	int <b>Value:492</b>
Constants.FP_FirstSelectedComponentIn Book	int <b>Value:493</b>
Constants.FP_BookDontUpdateReferences	int <b>Value:494</b>
Constants.FP_BooksViewOnly	int <b>Value:497</b>
Constants.FP_BooksSelected	int <b>Value:499</b>
Constants.FP_TypeOfDisplayText	int <b>Value:500</b>

Property name	Data Type
Constants.FV_BK_FILENAME	int <b>Value:1</b>
Constants.FV_BK_TEXT	int <b>Value:2</b>
Constants.FP_BookParent	int <b>Value:515</b>
Constants.FP_ExtractTags	int <b>Value:516</b>
Constants.FP_GenerateInclude	int <b>Value:517</b>
Constants.FP_ImportFmtInclude	int <b>Value:518</b>
Constants.FP_PrintInclude	int <b>Value:519</b>
Constants.FP_BookComponentIsGeneratable	int <b>Value:520</b>
Constants.FP_PagePrefix	int <b>Value:521</b>
Constants.FP_PageSuffix	int <b>Value:522</b>
Constants.FP_PageSide	int <b>Value:523</b>
Constants.FV_BK_START_FROM_FILE	int <b>Value:0x01</b>
Constants.FV_BK_START_NEXT_AVAILABLE	int <b>Value:0x02</b>
Constants.FV_BK_START_LEFT	int <b>Value:0x03</b>
Constants.FV_BK_START_RIGHT	int <b>Value:0x04</b>
Constants.FP_PageNumComputeMethod	int <b>Value:524</b>
Constants.FP_PgfNumComputeMethod	int <b>Value:525</b>

Property name	Data Type
Constants.FP_PrevComponentInBook	int <b>Value:526</b>
Constants.FP_NextComponentInBook	int <b>Value:527</b>
Constants.FP_InsertLinks	int <b>Value:528</b>
Constants.FP_ComponentIsSelected	int <b>Value:529</b>
Constants.FP_NextSelectedComponentInBook	int <b>Value:530</b>
Constants.FP_BookComponentType	int <b>Value:531</b>
Constants.FV_BK_TOC	int <b>Value:0</b>
Constants.FV_BK_LIST_FIGURE	int <b>Value:1</b>
Constants.FV_BK_LIST_TABLE	int <b>Value:2</b>
Constants.FV_BK_LIST_PGF	int <b>Value:3</b>
Constants.FV_BK_LIST_MARKER	int <b>Value:4</b>
Constants.FV_BK_LIST_MARKER_ALPHA	int <b>Value:5</b>
Constants.FV_BK_LIST_PGF_ALPHA	int <b>Value:6</b>
Constants.FV_BK_INDEX_STAN	int <b>Value:7</b>
Constants.FV_BK_INDEX_AUTHOR	int <b>Value:8</b>
Constants.FV_BK_INDEX_SUBJECT	int <b>Value:9</b>
Constants.FV_BK_INDEX_MARKER	int <b>Value:10</b>

Property name	Data Type
Constants.FV_BK_LIST_FORMATS	int <b>Value:11</b>
Constants.FV_BK_LIST_REFERENCES	int <b>Value:12</b>
Constants.FV_BK_INDEX_FORMATS	int <b>Value:13</b>
Constants.FV_BK_INDEX_REFERENCES	int <b>Value:14</b>
Constants.FV_BK_NUM_BOOK_LIST	int <b>Value:15</b>
Constants.FV_BK_NOT_GENERATABLE	int <b>Value:16</b>
Constants.FP_ComponentDisplayText	int <b>Value:532</b>
Constants.FP_FirstComponentInBookComponent	int <b>Value:2324</b>
Constants.FP_BookComponentParent	int <b>Value:2325</b>
Constants.FP_ExcludeBookComponent	int <b>Value:2326</b>
Constants.FP_BookComponentTemplatePath	int <b>Value:2327</b>
Constants.FP_BookComponentTitle	int <b>Value:2328</b>
Constants.FP_ComponentType	int <b>Value:2329</b>
Constants.FP_ComponentIsDitaMap	int <b>Value:2331</b>
Constants.FP_CorrespondingXmlPath	int <b>Value:1158</b>
Constants.FV_BK_GENERAL	int <b>Value:0x1</b>
Constants.FV_BK_FOLDER	int <b>Value:0x2</b>



Property name	Data Type
Constants.FV_BK_BOOK	int <b>Value:</b> 0x4
Constants.FV_BK_FM	int <b>Value:</b> 0x8
Constants.FV_BK_MIF	int <b>Value:</b> 0x10
Constants.FV_BK_XML	int <b>Value:</b> 0x20
Constants.FV_BK_GROUP	int <b>Value:</b> 0x40
Constants.FV_BK_DITAMAP	int <b>Value:</b> 0x80
Constants.FV_BK_BOOKMAP	int <b>Value:</b> 0x100
Constants.FV_BK_FILE	int <b>Value:</b> 0x200
Constants.FP_NextBookComponentInDF SOrder	int <b>Value:</b> 2333
Constants.FP_BookComponentFileType	int <b>Value:</b> 2339
Constants.FP_PrevBookComponentInDF SOrder	int <b>Value:</b> 2340
Constants.FP_BookComponentIsFolder WithTemplate	int <b>Value:</b> 2341
Constants.FP_BookComponentIsFolder WithoutTemplate	int <b>Value:</b> 2342
Constants.FP_XmlApplicationForBookCo mponent	int <b>Value:</b> 2330
Constants.FP_BookComponentSequence Level	int <b>Value:</b> 2353
Constants.FP_DocSequenceLevel	int <b>Value:</b> 2356
Constants.FA_COMPONENT_MOVEUP	int <b>Value:</b> 1

Property name	Data Type
Constants.FA_COMPONENT_MOVEDOWN	int <b>Value:2</b>
Constants.FA_COMPONENT_PROMOTE	int <b>Value:3</b>
Constants.FA_COMPONENT_DEMOTE	int <b>Value:4</b>
Constants.FP_PgfAlignment	int <b>Value:547</b>
Constants.FV_PGF_LEFT	int <b>Value:0x01</b>
Constants.FV_PGF_RIGHT	int <b>Value:0x02</b>
Constants.FV_PGF_CENTER	int <b>Value:0x03</b>
Constants.FV_PGF_JUSTIFIED	int <b>Value:0x04</b>
Constants.FP_AutoNumString	int <b>Value:548</b>
Constants.FP_AutoNumChar	int <b>Value:549</b>
Constants.FP_LetterSpace	int <b>Value:550</b>
Constants.FP_KeepWithPrev	int <b>Value:551</b>
Constants.FP_NextTag	int <b>Value:552</b>
Constants.FP_SpaceAbove	int <b>Value:553</b>
Constants.FP_TopSeparator	int <b>Value:554</b>
Constants.FP_LeftIndent	int <b>Value:555</b>
Constants.FirstInlineComponentInDoc	int <b>Value:2897</b>

Property name	Data Type
Constants.FP_OptSpace	int <b>Value:</b> 557
Constants.FP_Leading	int <b>Value:</b> 558
Constants.FP_AdjHyphens	int <b>Value:</b> 559
Constants.FP_HyphMinSuffix	int <b>Value:</b> 560
Constants.FP_BlockLines	int <b>Value:</b> 561
Constants.FP_PgflsAutoNum	int <b>Value:</b> 562
Constants.FP_NumAtEnd	int <b>Value:</b> 563
Constants.FP_Hyphenate	int <b>Value:</b> 564
Constants.FP_KeepWithNext	int <b>Value:</b> 565
Constants.FP_UseNextTag	int <b>Value:</b> 566
Constants.FP_Start	int <b>Value:</b> 567
Constants.FV_PGF_ANYWHERE	int <b>Value:</b> 0x00
Constants.FV_PGF_TOP_OF_COL	int <b>Value:</b> 0x01
Constants.FV_PGF_TOP_OF_PAGE	int <b>Value:</b> 0x02
Constants.FV_PGF_TOP_OF_LEFT_PAGE	int <b>Value:</b> 0x03
Constants.FV_PGF_TOP_OF_RIGHT_PAGE	int <b>Value:</b> 0x04
Constants.FP_SpaceBelow	int <b>Value:</b> 568

Property name	Data Type
Constants.FP_BottomSeparator	int <b>Value:</b> 569
Constants.FP_RightIndent	int <b>Value:</b> 570
Constants.FP_MinSpace	int <b>Value:</b> 571
Constants.FP_MaxSpace	int <b>Value:</b> 572
Constants.FP_NumTabs	int <b>Value:</b> 573
Constants.FP_Tabs	int <b>Value:</b> 574
Constants.FP_HyphMinPrefix	int <b>Value:</b> 575
Constants.FP_HyphMinWord	int <b>Value:</b> 576
Constants.FP_Language	int <b>Value:</b> 577
Constants.FV_LANG_NOLANGUAGE	int <b>Value:</b> 0x00
Constants.FV_LANG_ENGLISH	int <b>Value:</b> 0x01
Constants.FV_LANG_BRITISH	int <b>Value:</b> 0x02
Constants.FV_LANG_GERMAN	int <b>Value:</b> 0x03
Constants.FV_LANG_SWISS_GERMAN	int <b>Value:</b> 0x04
Constants.FV_LANG_FRENCH	int <b>Value:</b> 0x05
Constants.FV_LANG_CANADIAN_FRENCH	int <b>Value:</b> 0x06
Constants.FV_LANG_SPANISH	int <b>Value:</b> 0x07

Property name	Data Type
Constants.FV_LANG_CATALAN	int <b>Value:</b> 0x08
Constants.FV_LANG_ITALIAN	int <b>Value:</b> 0x09
Constants.FV_LANG_PORTUGUESE	int <b>Value:</b> 0x0A
Constants.FV_LANG_BRAZILIAN	int <b>Value:</b> 0x0B
Constants.FV_LANG_DANISH	int <b>Value:</b> 0x0C
Constants.FV_LANG_DUTCH	int <b>Value:</b> 0x0D
Constants.FV_LANG_NORWEGIAN	int <b>Value:</b> 0x0E
Constants.FV_LANG_NYNORSK	int <b>Value:</b> 0x0F
Constants.FV_LANG_FINNISH	int <b>Value:</b> 0x10
Constants.FV_LANG_SWEDISH	int <b>Value:</b> 0x11
Constants.FV_LANG_JAPANESE	int <b>Value:</b> 0x12
Constants.FV_LANG_TRADITIONAL_CHINESE	int <b>Value:</b> 0x13
Constants.FV_LANG_SIMPLIFIED_CHINESE	int <b>Value:</b> 0x14
Constants.FV_LANG_KOREAN	int <b>Value:</b> 0x15
Constants.FV_LANG_NEW_GERMAN	int <b>Value:</b> 0x16
Constants.FV_LANG_NEW_SWISS_GERMAN	int <b>Value:</b> 0x17
Constants.FV_LANG_NEW_DUTCH	int <b>Value:</b> 0x18

Property name	Data Type
Constants.FV_LANG_GREEK	int Value:0x19
Constants.FV_LANG_RUSSIAN	int Value:0x1A
Constants.FV_LANG_CZECH	int Value:0x1B
Constants.FV_LANG_POLISH	int Value:0x1C
Constants.FV_LANG_HUNGARIAN	int Value:0x1D
Constants.FV_LANG_TURKISH	int Value:0x1E
Constants.FV_LANG_SLOVAK	int Value:0x1F
Constants.FV_LANG_SLOVENIAN	int Value:0x20
Constants.FV_LANG_BULGARIAN	int Value:0x21
Constants.FV_LANG_CROATIAN	int Value:0x22
Constants.FV_LANG_ESTONIAN	int Value:0x23
Constants.FV_LANG_LATVIAN	int Value:0x24
Constants.FV_LANG_LITHUANIAN	int Value:0x25
Constants.FV_LANG_ROMANIAN	int Value:0x26
Constants.FV_LANG_NUM	int Value:0x27
Constants.FP_RunInSeparator	int Value:578
Constants.FP_Placement	int Value:579

Property name	Data Type
Constants.FV_PGF_SIDEBODY	int <b>Value:0</b>
Constants.FV_PGF_SIDEHEAD_TOP	int <b>Value:1</b>
Constants.FV_PGF_SIDEHEAD_FIRST_BAS SELINE	int <b>Value:2</b>
Constants.FV_PGF_SIDEHEAD_LAST_BAS ELINE	int <b>Value:3</b>
Constants.FV_PGF_RUN_IN	int <b>Value:4</b>
Constants.FV_PGF_STRADDLE	int <b>Value:5</b>
Constants.FV_PGF_STRADDLE_NORMAL _ONLY	int <b>Value:6</b>
Constants.FP_NextPgFmtInDoc	int <b>Value:580</b>
Constants.FP_CellTopMargin	int <b>Value:581</b>
Constants.FP_CellBottomMargin	int <b>Value:582</b>
Constants.FP_CellLeftMargin	int <b>Value:583</b>
Constants.FP_CellRightMargin	int <b>Value:584</b>
Constants.FP_CellVAlignment	int <b>Value:585</b>
Constants.FV_PGF_V_ALIGN_TOP	int <b>Value:0</b>
Constants.FV_PGF_V_ALIGN_MIDDLE	int <b>Value:1</b>
Constants.FV_PGF_V_ALIGN_BOTTOM	int <b>Value:2</b>
Constants.FP_CellMarginsFixed	int <b>Value:586</b>

Property name	Data Type
Constants.FV_PGF_FIXED_L_MARGIN	int <b>Value:</b> 0x01
Constants.FV_PGF_FIXED_B_MARGIN	int <b>Value:</b> 0x02
Constants.FV_PGF_FIXED_R_MARGIN	int <b>Value:</b> 0x04
Constants.FV_PGF_FIXED_T_MARGIN	int <b>Value:</b> 0x08
Constants.FP_LineSpacing	int <b>Value:</b> 587
Constants.FV_PGF_FIXED	int <b>Value:</b> 0x00
Constants.FV_PGF_PROPORTIONAL	int <b>Value:</b> 0x01
Constants.FV_PGF_FLOATING	int <b>Value:</b> 0x02
Constants.FP_Locked	int <b>Value:</b> 588
Constants.FP_AcrobatLevel	int <b>Value:</b> 589
Constants.FP_PDFStructureLevel	int <b>Value:</b> 600
Constants.FP_FormatOverride	int <b>Value:</b> 590
Constants.FP_MinJRomSpace	int <b>Value:</b> 591
Constants.FP_OptJRomSpace	int <b>Value:</b> 592
Constants.FP_MaxJRomSpace	int <b>Value:</b> 593
Constants.FP_MinJLetSpace	int <b>Value:</b> 594
Constants.FP_OptJLetSpace	int <b>Value:</b> 595



Property name	Data Type
Constants.FP_MaxJLetSpace	int <b>Value:</b> 596
Constants.FP_YakumonoType	int <b>Value:</b> 597
Constants.FV_FLOATING_YAKUMONO	int <b>Value:</b> 0x00
Constants.FV_MONOSPACE_YAKUMONO	int <b>Value:</b> 0x01
Constants.FV_FIXED_YAKUMONO	int <b>Value:</b> 0x02
Constants.FP_DialogEncodingName	int <b>Value:</b> 598
Constants.FP_FMInterfaceEncodingName	int <b>Value:</b> 599
Constants.FP_CharTag	int <b>Value:</b> 602
Constants.FP_NextCharFmtInDoc	int <b>Value:</b> 603
Constants.FP_FontFamily	int <b>Value:</b> 604
Constants.FP_FontVariation	int <b>Value:</b> 605
Constants.FP_FontWeight	int <b>Value:</b> 606
Constants.FP_FontAngle	int <b>Value:</b> 607
Constants.FP_Underlining	int <b>Value:</b> 608
Constants.FV_CB_NO_UNDERLINE	int <b>Value:</b> 0
Constants.FV_CB_SINGLE_UNDERLINE	int <b>Value:</b> 1
Constants.FV_CB_DOUBLE_UNDERLINE	int <b>Value:</b> 2

Property name	Data Type
Constants.FV_CB_NUMERIC_UNDERLINE	int <b>Value:3</b>
Constants.FP_Strikethrough	int <b>Value:609</b>
Constants.FP_Overline	int <b>Value:610</b>
Constants.FP_ChangeBar	int <b>Value:611</b>
Constants.FP_Outline	int <b>Value:612</b>
Constants.FP_Shadow	int <b>Value:613</b>
Constants.FP_PairKern	int <b>Value:614</b>
Constants.FP_FontSize	int <b>Value:615</b>
Constants.FP_KernX	int <b>Value:616</b>
Constants.FP_KernY	int <b>Value:617</b>
Constants.FP_Spread	int <b>Value:618</b>
Constants.FP_Capitalization	int <b>Value:619</b>
Constants.FV_CAPITAL_CASE_NORM	int <b>Value:0</b>
Constants.FV_CAPITAL_CASE_SMALL	int <b>Value:1</b>
Constants.FV_CAPITAL_CASE_LOWER	int <b>Value:2</b>
Constants.FV_CAPITAL_CASE_UPPER	int <b>Value:3</b>
Constants.FP_Position	int <b>Value:620</b>

Property name	Data Type
Constants.FV_POS_NORM	int <b>Value:0</b>
Constants.FV_POS_SUPER	int <b>Value:1</b>
Constants.FV_POS_SUB	int <b>Value:2</b>
Constants.FP_BkColor	int <b>Value:992</b>
Constants.FP_ReviewerName	int <b>Value:993</b>
Constants.FP_ReviewTime	int <b>Value:994</b>
Constants.FP_UseFontFamily	int <b>Value:621</b>
Constants.FP_UseFontVariation	int <b>Value:622</b>
Constants.FP_UseFontWeight	int <b>Value:623</b>
Constants.FP_UseFontAngle	int <b>Value:624</b>
Constants.FP_UseUnderlining	int <b>Value:625</b>
Constants.FP_UseStrikethrough	int <b>Value:626</b>
Constants.FP_UseOverline	int <b>Value:627</b>
Constants.FP_UseChangeBar	int <b>Value:628</b>
Constants.FP_UseOutline	int <b>Value:629</b>
Constants.FP_UseShadow	int <b>Value:630</b>
Constants.FP_UsePairKern	int <b>Value:631</b>

Property name	Data Type
Constants.FP_UseFontSize	int <b>Value:632</b>
Constants.FP_UseKernX	int <b>Value:633</b>
Constants.FP_UseKernY	int <b>Value:634</b>
Constants.FP_UseSpread	int <b>Value:635</b>
Constants.FP_UseCapitalization	int <b>Value:636</b>
Constants.FP_UsePosition	int <b>Value:637</b>
Constants.FP_UseColor	int <b>Value:638</b>
Constants.FP_UseBkColor	int <b>Value:991</b>
Constants.FP_FontPlatformName	int <b>Value:639</b>
Constants.FP_FontPostScriptName	int <b>Value:640</b>
Constants.FP_FontPanoseName	int <b>Value:641</b>
Constants.FP_FontEncodingName	int <b>Value:642</b>
Constants.FP_Stretch	int <b>Value:643</b>
Constants.FP_UseStretch	int <b>Value:644</b>
Constants.FP_UseLanguage	int <b>Value:645</b>
Constants.FP_WesternFontPlatformName	int <b>Value:646</b>
Constants.FP_WesternFontPostScriptName	int <b>Value:647</b>

Property name	Data Type
Constants.FP_WesternFontPanoseName	int Value:648
Constants.FP_Tsume	int Value:649
Constants.FP_UseTsume	int Value:650
Constants.FV_TAB_LEFT	int Value:0x1
Constants.FV_TAB_CENTER	int Value:0x2
Constants.FV_TAB_RIGHT	int Value:0x3
Constants.FV_TAB_DECIMAL	int Value:0x4
Constants.FV_TAB_RELATIVE_LEFT	int Value:0x5
Constants.FV_TAB_RELATIVE_CENTER	int Value:0x6
Constants.FV_TAB_RELATIVE_RIGHT	int Value:0x7
Constants.FV_TAB_RELATIVE_DECIMAL	int Value:0x8
Constants.FP_PageNum	int Value:687
Constants.FP_PointPageNum	int Value:688
Constants.FP_PageBackground	int Value:689
Constants.FV_BGD_DEFAULT	int Value:0x00
Constants.FV_BGD_NONE	int Value:0x01
Constants.FV_BGD_OTHER	int Value:0x02

Property name	Data Type
Constants.FP_MasterPage	int Value:690
Constants.FP_PageWidth	int Value:691
Constants.FP_PageHeight	int Value:692
Constants.FP_PagePrev	int Value:693
Constants.FP_PageNext	int Value:694
Constants.FP_PageFrame	int Value:695
Constants.FP_PageNumString	int Value:696
Constants.FP_PageIsRecto	int Value:697
Constants.FP_NextPgInDoc	int Value:716
Constants.FP_NextPgInFlow	int Value:717
Constants.FP_PrevPgInFlow	int Value:718
Constants.FP_PgfNumber	int Value:719
Constants.FP_PgfSpellChecked	int Value:720
Constants.FP_PgfSplit	int Value:721
Constants.FP_PgfMarkedForNamedDestination	int Value:722
Constants.FP_BorderWidth	int Value:740
Constants.FP_Fill	int Value:741

Property name	Data Type
Constants.FV_FILL_BLACK	int <b>Value:0</b>
Constants.FV_FILL_WHITE	int <b>Value:7</b>
Constants.FV_FILL_CLEAR	int <b>Value:15</b>
Constants.FP_Pen	int <b>Value:742</b>
Constants.FP_HeadArrow	int <b>Value:743</b>
Constants.FP_TailArrow	int <b>Value:744</b>
Constants.FP_ArrowTipAngle	int <b>Value:745</b>
Constants.FP_ArrowBaseAngle	int <b>Value:746</b>
Constants.FP_ArrowScaleFactor	int <b>Value:747</b>
Constants.FP_ArrowLength	int <b>Value:748</b>
Constants.FP_ArrowType	int <b>Value:749</b>
Constants.FV_ARROW_STICK	int <b>Value:0x1</b>
Constants.FV_ARROW_HOLLOW	int <b>Value:0x2</b>
Constants.FV_ARROW_FILLED	int <b>Value:0x3</b>
Constants.FP_ArrowScaleHead	int <b>Value:750</b>
Constants.FP_Color	int <b>Value:751</b>
Constants.FP_Dash	int <b>Value:752</b>

Property name	Data Type
Constants.FP_LineCap	int <b>Value:</b> 753
Constants.FV_CAP_BUTT	int <b>Value:</b> 0x00
Constants.FV_CAP_ROUND	int <b>Value:</b> 0x01
Constants.FV_CAP_SQUARE	int <b>Value:</b> 0x02
Constants.FP_RunaroundGap	int <b>Value:</b> 754
Constants.FP_TintPercent	int <b>Value:</b> 755
Constants.FP_Overprint	int <b>Value:</b> 756
Constants.FV_KNOCKOUT	int <b>Value:</b> 0x00
Constants.FV_OVERPRINT	int <b>Value:</b> 0x01
Constants.FV_FROMCOLOR	int <b>Value:</b> 0x02
Constants.FP_UseBorderWidth	int <b>Value:</b> 3001
Constants.FP_UseFill	int <b>Value:</b> 3002
Constants.FP_UsePen	int <b>Value:</b> 3003
Constants.FP_UseRunaroundGap	int <b>Value:</b> 3004
Constants.FP_UseTintPercent	int <b>Value:</b> 3005
Constants.FP_UseOverprint	int <b>Value:</b> 3006
Constants.FP_UseAngle	int <b>Value:</b> 3007



Property name	Data Type
Constants.FP_UseLocX	int <b>Value:3008</b>
Constants.FP_UseLocY	int <b>Value:3009</b>
Constants.FP_UseWidth	int <b>Value:3010</b>
Constants.FP_UseHeight	int <b>Value:3011</b>
Constants.FP_UseRunaround	int <b>Value:3012</b>
Constants.FP_UseAnchorType	int <b>Value:3013</b>
Constants.FP_UseAFramelsFloating	int <b>Value:3014</b>
Constants.FP_UseAFramelsCropped	int <b>Value:3015</b>
Constants.FP_UseSideOffset	int <b>Value:3016</b>
Constants.FP_UseBaselineOffset	int <b>Value:3017</b>
Constants.FP_UseAlignment	int <b>Value:3018</b>
Constants.FP_UseNumColumns	int <b>Value:3019</b>
Constants.FP_UseColGapWidth	int <b>Value:3020</b>
Constants.FP_UseSideHeadWidth	int <b>Value:3021</b>
Constants.FP_UseSideHeadGap	int <b>Value:3022</b>
Constants.FP_UseTheta	int <b>Value:3023</b>
Constants.FP_UseDTheta	int <b>Value:3024</b>

Property name	Data Type
Constants.FP_UseTextLineType	int <b>Value:</b> 3025
Constants.FP_UseMathSize	int <b>Value:</b> 3026
Constants.FP_UseInsetDpi	int <b>Value:</b> 3027
Constants.FP_UseColumnsAreBalanced	int <b>Value:</b> 3028
Constants.FP_UseSideHeadPlacement	int <b>Value:</b> 3029
Constants.FP_UseRadius	int <b>Value:</b> 3030
Constants.FP_UseFlowsAutoConnect	int <b>Value:</b> 3031
Constants.FP_UseFlowsPostScript	int <b>Value:</b> 3032
Constants.FP_FirstGraphicsFmtInDoc	int <b>Value:</b> 3101
Constants.FP_NextGraphicsFmtInDoc	int <b>Value:</b> 3102
Constants.FP_StyleTag	int <b>Value:</b> 770
Constants.FP_GraphicsSelected	int <b>Value:</b> 771
Constants.FP_GraphicCantBeSelected	int <b>Value:</b> 772
Constants.FP_GraphicsButton	int <b>Value:</b> 773
Constants.FP_FrameParent	int <b>Value:</b> 774
Constants.FP_PrevGraphicInFrame	int <b>Value:</b> 775
Constants.FP_NextGraphicInFrame	int <b>Value:</b> 776

Property name	Data Type
Constants.FP_GroupParent	int <b>Value:777</b>
Constants.FP_PrevGraphicInGroup	int <b>Value:778</b>
Constants.FP_NextGraphicInGroup	int <b>Value:779</b>
Constants.FP_Angle	int <b>Value:780</b>
Constants.FP_LocX	int <b>Value:781</b>
Constants.FP_LocY	int <b>Value:782</b>
Constants.FP_Width	int <b>Value:783</b>
Constants.FP_Height	int <b>Value:784</b>
Constants.FP_NextSelectedGraphicInDoc	int <b>Value:785</b>
Constants.FP_NextGraphicInDoc	int <b>Value:786</b>
Constants.FP_Runaround	int <b>Value:787</b>
Constants.FV_TR_NONE	int <b>Value:0x01</b>
Constants.FV_TR_CONTOUR	int <b>Value:0x02</b>
Constants.FV_TR_BBOX	int <b>Value:0x03</b>
Constants.FP_DesktopX	int <b>Value:788</b>
Constants.FP_DesktopY	int <b>Value:789</b>
Constants.FP_DesktopWidth	int <b>Value:790</b>

Property name	Data Type
Constants.FP_DesktopHeight	int <b>Value:791</b>
Constants.FP_IsHotspot	int <b>Value:2875</b>
Constants.FP_HotspotCmdStr	int <b>Value:2876</b>
Constants.FP_HotspotTitle	int <b>Value:2877</b>
Constants.FP_FirstGraphicInGroup	int <b>Value:807</b>
Constants.FP_LastGraphicInGroup	int <b>Value:808</b>
Constants.FP_DTheta	int <b>Value:830</b>
Constants.FP_Theta	int <b>Value:831</b>
Constants.FP_InsetEditor	int <b>Value:850</b>
Constants.FP_InsetUpdater	int <b>Value:851</b>
Constants.FP_InsetFile	int <b>Value:852</b>
Constants.FP_InsetFileOrigName	int <b>Value:2814</b>
Constants.FP_InsetDpi	int <b>Value:853</b>
Constants.FP_InsetIsFixedSize	int <b>Value:854</b>
Constants.FP_InsetIsFlippedSideways	int <b>Value:855</b>
Constants.FP_InsetIsInverted	int <b>Value:856</b>
Constants.FP_InsetRasterDpi	int <b>Value:857</b>

Property name	Data Type
Constants.FP_INSETinfo	int <b>Value:</b> 858
Constants.FP_InsetPosterIsExternal	int <b>Value:</b> 2821
Constants.FP_InsetPosterFile	int <b>Value:</b> 2822
Constants.FP_InsetSaveDIBFacetToFile	int <b>Value:</b> 2823
Constants.FP_InsetPosterFileOrigName	int <b>Value:</b> 2824
Constants.FP_InsetSaveFacetToFile	int <b>Value:</b> 2839
Constants.FP_InsetU3dAnimationList	int <b>Value:</b> 2844
Constants.FP_InsetU3dPartList	int <b>Value:</b> 2845
Constants.FP_InsetU3dViewList	int <b>Value:</b> 2846
Constants.FP_InsetCuePointList	int <b>Value:</b> 2354
Constants.FP_InsetLinkToText	int <b>Value:</b> 2355
Constants.FP_MathFullForm	int <b>Value:</b> 865
Constants.FP_MathSize	int <b>Value:</b> 866
Constants.FV_MATH_MEDIUM	int <b>Value:</b> 0
Constants.FV_MATH_SMALL	int <b>Value:</b> 1
Constants.FV_MATH_LARGE	int <b>Value:</b> 2
Constants.FP_AnchorType	int <b>Value:</b> 875

Property name	Data Type
Constants.FV_ANCHOR_INLINE	int <b>Value:1</b>
Constants.FV_ANCHOR_TOP	int <b>Value:2</b>
Constants.FV_ANCHOR_BELOW	int <b>Value:3</b>
Constants.FV_ANCHOR_BOTTOM	int <b>Value:4</b>
Constants.FV_ANCHOR_SUBCOL_LEFT	int <b>Value:5</b>
Constants.FV_ANCHOR_SUBCOL_RIGHT	int <b>Value:6</b>
Constants.FV_ANCHOR_SUBCOL_NEAREST	int <b>Value:7</b>
Constants.FV_ANCHOR_SUBCOL_FARTHEST	int <b>Value:8</b>
Constants.FV_ANCHOR_SUBCOL_INSIDE	int <b>Value:9</b>
Constants.FV_ANCHOR_SUBCOL_OUTSIDE	int <b>Value:10</b>
Constants.FV_ANCHOR_TEXTFRAME_LEFT	int <b>Value:11</b>
Constants.FV_ANCHOR_TEXTFRAME_RIGHT	int <b>Value:12</b>
Constants.FV_ANCHOR_TEXTFRAME_NEAREST	int <b>Value:13</b>
Constants.FV_ANCHOR_TEXTFRAME_FARTHEST	int <b>Value:14</b>
Constants.FV_ANCHOR_TEXTFRAME_INSIDE	int <b>Value:15</b>
Constants.FV_ANCHOR_TEXTFRAME_OUTSIDE	int <b>Value:16</b>
Constants.FV_ANCHOR_RUN_INTO_PARAGRAPH	int <b>Value:17</b>

Property name	Data Type
Constants.FP_AFrameIsFloating	int <b>Value:876</b>
Constants.FP_SideOffset	int <b>Value:877</b>
Constants.FP_AFrameIsCropped	int <b>Value:878</b>
Constants.FP_TextLoc	int <b>Value:879</b>
Constants.FP_PageFramePage	int <b>Value:880</b>
Constants.FP_BaselineOffset	int <b>Value:881</b>
Constants.FP_FirstGraphicInFrame	int <b>Value:882</b>
Constants.FP_LastGraphicInFrame	int <b>Value:883</b>
Constants.FP_PrevAFrame	int <b>Value:884</b>
Constants.FP_NextAFrame	int <b>Value:885</b>
Constants.FP_Alignment	int <b>Value:886</b>
Constants.FV_ALIGN_LEFT	int <b>Value:0</b>
Constants.FV_ALIGN_CENTER	int <b>Value:1</b>
Constants.FV_ALIGN_RIGHT	int <b>Value:2</b>
Constants.FV_ALIGN_INSIDE	int <b>Value:3</b>
Constants.FV_ALIGN_OUTSIDE	int <b>Value:4</b>
Constants.FP_MifFileEntity	int <b>Value:2828</b>

Property name	Data Type
Constants.FP_MifFileName	int <b>Value:2829</b>
Constants.FP_MifFileOrigFormat	int <b>Value:2878</b>
Constants.FP_PolyIsBezier	int <b>Value:906</b>
Constants.FP_NumPoints	int <b>Value:907</b>
Constants.FP_Points	int <b>Value:908</b>
Constants.FP_RectangleIsSmoothed	int <b>Value:929</b>
Constants.FP_Radius	int <b>Value:950</b>
Constants.FP_FirstPgf	int <b>Value:972</b>
Constants.FP_LastPgf	int <b>Value:973</b>
Constants.FP_FirstAFrame	int <b>Value:974</b>
Constants.FP_LastAFrame	int <b>Value:975</b>
Constants.FP_FirstFn	int <b>Value:976</b>
Constants.FP_LastFn	int <b>Value:977</b>
Constants.FP_FirstCell	int <b>Value:978</b>
Constants.FP_LastCell	int <b>Value:979</b>
Constants.FP_PrevTextFrameInFlow	int <b>Value:980</b>
Constants.FP_NextTextFrameInFlow	int <b>Value:981</b>



Property name	Data Type
Constants.FP_Flow	int <b>Value:982</b>
Constants.FP_NumColumns	int <b>Value:984</b>
Constants.FP_ColGapWidth	int <b>Value:985</b>
Constants.FP_FirstSubCol	int <b>Value:986</b>
Constants.FP_LastSubCol	int <b>Value:987</b>
Constants.FP_LineSpacingFactor	int <b>Value:988</b>
Constants.FP_PgfSpacingFactor	int <b>Value:989</b>
Constants.FP_CellMarginSpacingDelta	int <b>Value:990</b>
Constants.FP_SideHeadPlacement	int <b>Value:1171</b>
Constants.FV_SH_LEFT	int <b>Value:0x00</b>
Constants.FV_SH_RIGHT	int <b>Value:0x01</b>
Constants.FV_SH_INSIDE	int <b>Value:0x02</b>
Constants.FV_SH_OUTSIDE	int <b>Value:0x03</b>
Constants.FP_SideHeadWidth	int <b>Value:1172</b>
Constants.FP_SideHeadGap	int <b>Value:1173</b>
Constants.FP_ColumnsAreBalanced	int <b>Value:1175</b>
Constants.FP_InTextObj	int <b>Value:1000</b>

Property name	Data Type
Constants.FP_InTextFrame	int <b>Value:</b> 1001
Constants.FP_FnNum	int <b>Value:</b> 1002
Constants.FP_NextFnInDoc	int <b>Value:</b> 1003
Constants.FP_PrevFn	int <b>Value:</b> 1004
Constants.FP_NextFn	int <b>Value:</b> 1005
Constants.FP_FnAnchorString	int <b>Value:</b> 1006
Constants.FP_OldTypeNum	int <b>Value:</b> 1024
Constants.FP_MarkerText	int <b>Value:</b> 1025
Constants.FP_NextMarkerInDoc	int <b>Value:</b> 1026
Constants.FP_MarkerTypeId	int <b>Value:</b> 1027
Constants.FP_NextMarkerTypeIdInDoc	int <b>Value:</b> 1028
Constants.FP_InvariantName	int <b>Value:</b> 1029
Constants.FP_Public	int <b>Value:</b> 1030
Constants.FP_Transient	int <b>Value:</b> 1031
Constants.FP_Required	int <b>Value:</b> 1032
Constants.FP_NextVarInDoc	int <b>Value:</b> 1046
Constants.FP_VarFmt	int <b>Value:</b> 1047

Property name	Data Type
Constants.FP_SystemVar	int <b>Value:1049</b>
Constants.FV_VAR_USER_VARIABLE	int <b>Value:0</b>
Constants.FV_VAR_CURRENT_PAGE_NUM	int <b>Value:1</b>
Constants.FV_VAR_PAGE_COUNT	int <b>Value:2</b>
Constants.FV_VAR_CURRENT_DATE_LONG	int <b>Value:3</b>
Constants.FV_VAR_CURRENT_DATE_SHORT	int <b>Value:4</b>
Constants.FV_VAR_MODIFICATION_DATE_LONG	int <b>Value:5</b>
Constants.FV_VAR_MODIFICATION_DATE_SHORT	int <b>Value:6</b>
Constants.FV_VAR_CREATION_DATE_LONG	int <b>Value:7</b>
Constants.FV_VAR_CREATION_DATE_SHORT	int <b>Value:8</b>
Constants.FV_VAR_FILE_NAME_LONG	int <b>Value:9</b>
Constants.FV_VAR_FILE_NAME_SHORT	int <b>Value:10</b>
Constants.FV_VAR_HEADER_FOOTER_1	int <b>Value:11</b>
Constants.FV_VAR_HEADER_FOOTER_2	int <b>Value:12</b>
Constants.FV_VAR_HEADER_FOOTER_3	int <b>Value:13</b>
Constants.FV_VAR_HEADER_FOOTER_4	int <b>Value:14</b>
Constants.FV_VAR_TABLE_CONTINUATION	int <b>Value:15</b>

Property name	Data Type
Constants.FV_VAR_TABLE_SHEET	int <b>Value:16</b>
Constants.FV_VAR_HEADER_FOOTER_5	int <b>Value:19</b>
Constants.FV_VAR_HEADER_FOOTER_6	int <b>Value:20</b>
Constants.FV_VAR_HEADER_FOOTER_7	int <b>Value:21</b>
Constants.FV_VAR_HEADER_FOOTER_8	int <b>Value:22</b>
Constants.FV_VAR_HEADER_FOOTER_9	int <b>Value:23</b>
Constants.FV_VAR_HEADER_FOOTER_10	int <b>Value:24</b>
Constants.FV_VAR_HEADER_FOOTER_11	int <b>Value:25</b>
Constants.FV_VAR_HEADER_FOOTER_12	int <b>Value:26</b>
Constants.FV_VAR_SECTION_NUMBER	int <b>Value:27</b>
Constants.FV_VAR_SUB_SECTION_NUMBER	int <b>Value:28</b>
Constants.FV_VAR_CHAPTER_TITLE	int <b>Value:29</b>
Constants.FV_VAR_USER_DIRECTORY	int <b>Value:30</b>
Constants.FV_VAR_HEADER_FOOTER_13	int <b>Value:31</b>
Constants.FV_VAR_HEADER_FOOTER_14	int <b>Value:32</b>
Constants.FV_VAR_HEADER_FOOTER_15	int <b>Value:33</b>
Constants.FV_VAR_HEADER_FOOTER_16	int <b>Value:34</b>

Property name	Data Type
Constants.FV_VAR_HEADER_FOOTER_17	int <b>Value:</b> 35
Constants.FV_VAR_HEADER_FOOTER_18	int <b>Value:</b> 36
Constants.FV_VAR_BOOK_TOTAL_PAGE_COUNT	int <b>Value:</b> 37
Constants.FP_Fmt	int <b>Value:</b> 1050
Constants.FP_NextVarFmtInDoc	int <b>Value:</b> 1051
Constants.FP_XRefFile	int <b>Value:</b> 1070
Constants.FP_TextRange	int <b>Value:</b> 1071
Constants.FP_NextXRefInDoc	int <b>Value:</b> 1072
Constants.FP_XRefFmt	int <b>Value:</b> 1073
Constants.FP_XRefSrcText	int <b>Value:</b> 1075
Constants.FP_XRefSrcsElem	int <b>Value:</b> 1076
Constants.FP_XRefsUnresolved	int <b>Value:</b> 1077
Constants.FP_XRefSrcElemNonUniqueId	int <b>Value:</b> 1152
Constants.FP_XRefAltText	int <b>Value:</b> 1153
Constants.FP_XRefClientName	int <b>Value:</b> 1154
Constants.FP_XRefClientType	int <b>Value:</b> 1155
Constants.FP_XRefResolvedWithKeyref	int <b>Value:</b> 1156

Property name	Data Type
Constants.FP_XRefRetainKeydefText	int <b>Value:1157</b>
Constants.FP_NextXRefFmtInDoc	int <b>Value:1074</b>
Constants.FP_TextLineType	int <b>Value:1095</b>
Constants.FV_TEXTLINE_LEFT	int <b>Value:0</b>
Constants.FV_TEXTLINE_RIGHT	int <b>Value:1</b>
Constants.FV_TEXTLINE_CENTER	int <b>Value:2</b>
Constants.FV_TEXTLINE_MATH	int <b>Value:3</b>
Constants.FP_BasePointX	int <b>Value:1116</b>
Constants.FP_BasePointY	int <b>Value:1117</b>
Constants.FP_NextCondFmtInDoc	int <b>Value:1138</b>
Constants.FP_CondFmtIsShown	int <b>Value:1139</b>
Constants.FP_SepOverride	int <b>Value:1140</b>
Constants.FP_UseSepOverride	int <b>Value:1141</b>
Constants.FP_StyleOverride	int <b>Value:1142</b>
Constants.FV_CN_NO_OVERRIDE	int <b>Value:0</b>
Constants.FV_CN_OVERLINE	int <b>Value:1</b>
Constants.FV_CN_STRIKETHROUGH	int <b>Value:2</b>

Property name	Data Type
Constants.FV_CN_SINGLE_UNDERLINE	int <b>Value:3</b>
Constants.FV_CN_DOUBLE_UNDERLINE	int <b>Value:4</b>
Constants.FV_CN_CHANGEBAR	int <b>Value:5</b>
Constants.FV_CN_NUMERIC_UNDERLINE	int <b>Value:6</b>
Constants.FV_CN_NMRIC_AND_CHNGBAR	int <b>Value:7</b>
Constants.FP_NextAttrCondExprInDoc	int <b>Value:2816</b>
Constants.FP_AttrCondExprStr	int <b>Value:2817</b>
Constants.FP_AttrCondExprIsActive	int <b>Value:2818</b>
Constants.FP_InCond	int <b>Value:1150</b>
Constants.FP_StyleOverrides	int <b>Value:1151</b>
Constants.FV_CS_NO_OVERRIDE	int <b>Value:0x00</b>
Constants.FV_CS_OVERLINE	int <b>Value:0x01</b>
Constants.FV_CS_STRIKETHROUGH	int <b>Value:0x02</b>
Constants.FV_CS_SINGLE_UNDERLINE	int <b>Value:0x04</b>
Constants.FV_CS_DOUBLE_UNDERLINE	int <b>Value:0x08</b>
Constants.FV_CS_CHANGEBAR	int <b>Value:0x10</b>
Constants.FV_CS_NUMERIC_UNDERLINE	int <b>Value:0x20</b>

Property name	Data Type
Constants.FP_NextFlowInDoc	int <b>Value:1162</b>
Constants.FP_FlowIsSynchronized	int <b>Value:1163</b>
Constants.FP_MinHang	int <b>Value:1164</b>
Constants.FP_FlowIsAutoConnect	int <b>Value:1165</b>
Constants.FP_FlowIsFeathered	int <b>Value:1166</b>
Constants.FP_Spacing	int <b>Value:1167</b>
Constants.FP_FlowIsPostScript	int <b>Value:1168</b>
Constants.FP_FirstTextFrameInFlow	int <b>Value:1169</b>
Constants.FP_LastTextFrameInFlow	int <b>Value:1170</b>
Constants.FP_SideHeadRoomInFlow	int <b>Value:1174</b>
Constants.FP_MaxInterlinePadding	int <b>Value:470</b>
Constants.FP_MaxInterPgfPadding	int <b>Value:471</b>
Constants.FP_CellUseOverrideLRuling	int <b>Value:1190</b>
Constants.FP_CellUseOverrideRRuling	int <b>Value:1191</b>
Constants.FP_CellUseOverrideTRuling	int <b>Value:1192</b>
Constants.FP_CellUseOverrideBRuling	int <b>Value:1193</b>
Constants.FP_CellOverrideShading	int <b>Value:1194</b>



Property name	Data Type
Constants.FP_CellOverrideFill	int <b>Value:1195</b>
Constants.FP_CellUseOverrideFill	int <b>Value:1196</b>
Constants.FP_CellUseOverrideShading	int <b>Value:1197</b>
Constants.FP_CellRow	int <b>Value:1198</b>
Constants.FP_NextCellInTbl	int <b>Value:1199</b>
Constants.FP_CellBelowInTbl	int <b>Value:1200</b>
Constants.FP_PrevCellInRow	int <b>Value:1201</b>
Constants.FP_NextCellInRow	int <b>Value:1202</b>
Constants.FP_CellAboveInCol	int <b>Value:1203</b>
Constants.FP_CellBelowInCol	int <b>Value:1204</b>
Constants.FP_CellColNum	int <b>Value:1205</b>
Constants.FP_CellsStraddled	int <b>Value:1206</b>
Constants.FP_CellNumRowsStraddled	int <b>Value:1207</b>
Constants.FP_CellNumColsStraddled	int <b>Value:1208</b>
Constants.FP_CellAngle	int <b>Value:1209</b>
Constants.FP_CellOverrideTopRuling	int <b>Value:1210</b>
Constants.FP_CellOverrideBottomRuling	int <b>Value:1211</b>

Property name	Data Type
Constants.FP_CellOverrideLeftRuling	int <b>Value:1212</b>
Constants.FP_CellOverrideRightRuling	int <b>Value:1213</b>
Constants.FP_CellDefaultTopRuling	int <b>Value:1214</b>
Constants.FP_CellDefaultBottomRuling	int <b>Value:1215</b>
Constants.FP_CellDefaultLeftRuling	int <b>Value:1216</b>
Constants.FP_CellDefaultRightRuling	int <b>Value:1217</b>
Constants.FP_CellsShown	int <b>Value:1218</b>
Constants.FP_PrevCell	int <b>Value:1219</b>
Constants.FP_NextCell	int <b>Value:1220</b>
Constants.FP_PrevRowInTbl	int <b>Value:1244</b>
Constants.FP_NextRowInTbl	int <b>Value:1245</b>
Constants.FP_RowTbl	int <b>Value:1246</b>
Constants.FP_FirstCellInRow	int <b>Value:1247</b>
Constants.FP_RowKeepWithNext	int <b>Value:1248</b>
Constants.FP_RowKeepWithPrev	int <b>Value:1249</b>
Constants.FP_RowMaxHeight	int <b>Value:1250</b>
Constants.FP_RowMinHeight	int <b>Value:1251</b>

Property name	Data Type
Constants.FP_RowStart	int Value:1252
Constants.FV_ROW_ANYWHERE	int Value:0
Constants.FV_ROW_TOP_OF_COL	int Value:1
Constants.FV_ROW_TOP_OF_PAGE	int Value:2
Constants.FV_ROW_TOP_OF_LEFT_PAGE	int Value:3
Constants.FV_ROW_TOP_OF_RIGHT_PAGE	int Value:4
Constants.FP_RowType	int Value:1253
Constants.FV_ROW_HEADING	int Value:0
Constants.FV_ROW_BODY	int Value:1
Constants.FV_ROW_FOOTING	int Value:2
Constants.FP_RowsShown	int Value:1254
Constants.FP_TblTag	int Value:1275
Constants.FP_NextTblFmtInDoc	int Value:1276
Constants.FP_TblLeftIndent	int Value:1278
Constants.FP_TblRightIndent	int Value:1279
Constants.FP_TblSpaceAbove	int Value:1280
Constants.FP_TblSpaceBelow	int Value:1281

Property name	Data Type
Constants.FP_TblAlignment	int <b>Value:1282</b>
Constants.FV_ALIGN_TBL_LEFT	int <b>Value:0</b>
Constants.FV_ALIGN_TBL_CENTER	int <b>Value:1</b>
Constants.FV_ALIGN_TBL_RIGHT	int <b>Value:2</b>
Constants.FP_TblPlacement	int <b>Value:1283</b>
Constants.FV_TBL_ANYWHERE	int <b>Value:0</b>
Constants.FV_TBL_TOP_OF_COL	int <b>Value:1</b>
Constants.FV_TBL_TOP_OF_PAGE	int <b>Value:2</b>
Constants.FV_TBL_TOP_OF_LEFT_PAGE	int <b>Value:3</b>
Constants.FV_TBL_TOP_OF_RIGHT_PAGE	int <b>Value:4</b>
Constants.FV_TBL_FLOAT	int <b>Value:5</b>
Constants.FP_TblInitNumCols	int <b>Value:1284</b>
Constants.FP_TblInitNumHRows	int <b>Value:1285</b>
Constants.FP_TblInitNumBodyRows	int <b>Value:1286</b>
Constants.FP_TblInitNumFRows	int <b>Value:1287</b>
Constants.FP_TblNumbering	int <b>Value:1288</b>
Constants.FV_TBL_NUM_BY_ROW	int <b>Value:0</b>

Property name	Data Type
Constants.FV_TBL_NUM_BY_COL	int <b>Value:1</b>
Constants.FP_TblTitlePosition	int <b>Value:1289</b>
Constants.FV_TBL_NO_TITLE	int <b>Value:0</b>
Constants.FV_TBL_TITLE_ABOVE	int <b>Value:1</b>
Constants.FV_TBL_TITLE_BELOW	int <b>Value:2</b>
Constants.FP_TblTitleGap	int <b>Value:1290</b>
Constants.FP_OrphanRows	int <b>Value:1291</b>
Constants.FP_TblCatalogEntry	int <b>Value:1292</b>
Constants.FP_TblColRulingPeriod	int <b>Value:1293</b>
Constants.FP_TblBodyRowRulingPeriod	int <b>Value:1294</b>
Constants.FP_TblLastBodyRuling	int <b>Value:1295</b>
Constants.FP_TblHFFill	int <b>Value:1296</b>
Constants.FP_TblHFColor	int <b>Value:1297</b>
Constants.FP_TblBodyFirstFill	int <b>Value:1298</b>
Constants.FP_TblBodyFirstColor	int <b>Value:1299</b>
Constants.FP_TblBodyShadeBy	int <b>Value:1300</b>
Constants.FP_TblBodyFirstPeriod	int <b>Value:1301</b>

Property name	Data Type
Constants.FP_TblBodyNextFill	int <b>Value:1302</b>
Constants.FP_TblBodyNextPeriod	int <b>Value:1303</b>
Constants.FP_TblBodyNextColor	int <b>Value:1304</b>
Constants.FP_TblTopRuling	int <b>Value:1305</b>
Constants.FP_TblBottomRuling	int <b>Value:1306</b>
Constants.FP_TblLeftRuling	int <b>Value:1307</b>
Constants.FP_TblRightRuling	int <b>Value:1308</b>
Constants.FP_TblColRuling	int <b>Value:1309</b>
Constants.FP_TblBodyRowRuling	int <b>Value:1310</b>
Constants.FP_TblHFSeparatorRuling	int <b>Value:1311</b>
Constants.FP_TblHFRowRuling	int <b>Value:1312</b>
Constants.FP_TblOtherBodyRowRuling	int <b>Value:1313</b>
Constants.FP_TblOtherColRuling	int <b>Value:1314</b>
Constants.FP_TblCellTopMargin	int <b>Value:1315</b>
Constants.FP_TblCellBottomMargin	int <b>Value:1316</b>
Constants.FP_TblCellLeftMargin	int <b>Value:1317</b>
Constants.FP_TblCellRightMargin	int <b>Value:1318</b>

Property name	Data Type
Constants.FP_TblInLockedTi	int <b>Value:1319</b>
Constants.FP_TblNumCols	int <b>Value:1335</b>
Constants.FP_TblNumRows	int <b>Value:1336</b>
Constants.FP_NextTblInDoc	int <b>Value:1337</b>
Constants.FP_FirstRowInTbl	int <b>Value:1338</b>
Constants.FP_LastRowInTbl	int <b>Value:1339</b>
Constants.FP_TblWidth	int <b>Value:1340</b>
Constants.FP_TopRowSelection	int <b>Value:1341</b>
Constants.FP_BottomRowSelection	int <b>Value:1342</b>
Constants.FP_LeftColNum	int <b>Value:1343</b>
Constants.FP_RightColNum	int <b>Value:1344</b>
Constants.FP_TblColWidths	int <b>Value:1345</b>
Constants.FP_TblTitleSelected	int <b>Value:1346</b>
Constants.FP_NextRulingFmtInDoc	int <b>Value:1430</b>
Constants.FP_RulingPenWidth	int <b>Value:1432</b>
Constants.FP_RulingGap	int <b>Value:1433</b>
Constants.FP_RulingSep	int <b>Value:1434</b>

Property name	Data Type
Constants.FP_RulingLines	int <b>Value:</b> 1435
Constants.FP_NextColorInDoc	int <b>Value:</b> 1455
Constants.FP_Pantone	int <b>Value:</b> 1456
Constants.FP_Cyan	int <b>Value:</b> 1457
Constants.FP_Magenta	int <b>Value:</b> 1458
Constants.FP_Yellow	int <b>Value:</b> 1459
Constants.FP_Black	int <b>Value:</b> 1460
Constants.FP_ColorViewCtl	int <b>Value:</b> 1461
Constants.FV_SEP_NORMAL	int <b>Value:</b> 0x0
Constants.FV_SEP_NONE	int <b>Value:</b> 0x1
Constants.FV_SEP_WHITE	int <b>Value:</b> 0x2
Constants.FP_ColorPrintCtl	int <b>Value:</b> 1462
Constants.FV_PRINT_SPOT	int <b>Value:</b> 0x0
Constants.FV_PRINT_PROCESS	int <b>Value:</b> 0x1
Constants.FV_PRINT_NO	int <b>Value:</b> 0x2
Constants.FP_ReservedColor	int <b>Value:</b> 1463
Constants.FV_COLOR_NOT_RESERVED	int <b>Value:</b> 0



Property name	Data Type
Constants.FV_COLOR_CYAN	int <b>Value:1</b>
Constants.FV_COLOR_MAGENTA	int <b>Value:2</b>
Constants.FV_COLOR_YELLOW	int <b>Value:3</b>
Constants.FV_COLOR_BLACK	int <b>Value:4</b>
Constants.FV_COLOR_WHITE	int <b>Value:5</b>
Constants.FV_COLOR_RED	int <b>Value:6</b>
Constants.FV_COLOR_GREEN	int <b>Value:7</b>
Constants.FV_COLOR_BLUE	int <b>Value:8</b>
Constants.FV_COLOR_DARKGREY	int <b>Value:9</b>
Constants.FV_COLOR_PALEGREEN	int <b>Value:10</b>
Constants.FV_COLOR_FORESTGREEN	int <b>Value:11</b>
Constants.FV_COLOR_ROYALBLUE	int <b>Value:12</b>
Constants.FV_COLOR_MAUVE	int <b>Value:13</b>
Constants.FV_COLOR_LIGHTSALMON	int <b>Value:14</b>
Constants.FV_COLOR_DARKYELLOW	int <b>Value:15</b>
Constants.FV_COLOR_SALMON	int <b>Value:16</b>
Constants.FP_FamilyName	int <b>Value:1464</b>

Property name	Data Type
Constants.FP_InkName	int <b>Value:</b> 1465
Constants.FP_ColorTintPercent	int <b>Value:</b> 1466
Constants.FP_TintBaseColor	int <b>Value:</b> 1467
Constants.FV_NO_BASE_COLOR	int <b>Value:</b> 0x00
Constants.FP_ColorOverprint	int <b>Value:</b> 1468
Constants.FV_COLOR_KNOCKOUT	int <b>Value:</b> 0x00
Constants.FV_COLOR_OVERPRINT	int <b>Value:</b> 0x01
Constants.FP_NextRubiInDoc	int <b>Value:</b> 1469
Constants.FP_OyamojiTextRange	int <b>Value:</b> 1470
Constants.FP_RubiTextRange	int <b>Value:</b> 1471
Constants.FP_HypertextDoValidate	int <b>Value:</b> 2300
Constants.FP_HypertextCommandText	int <b>Value:</b> 2301
Constants.FP_HypertextParsedArgs	int <b>Value:</b> 2302
Constants.FP_HypertextParseErr	int <b>Value:</b> 2303
Constants.FV_HypertextSyntaxOK	int <b>Value:</b> 0
Constants.FV_HypertextEmptyCommand	int <b>Value:</b> 1
Constants.FV_HypertextUnrecognizedCommand	int <b>Value:</b> 2

Property name	Data Type
Constants.FV_HypertextMissingArguments	int <b>Value:3</b>
Constants.FV_HypertextExtraArguments	int <b>Value:4</b>
Constants.FV_HypertextBadSyntaxPathSpec	int <b>Value:10</b>
Constants.FV_HypertextUnanchoredPartialPath	int <b>Value:11</b>
Constants.FV_HypertextHelpDirNotFound	int <b>Value:20</b>
Constants.FV_HypertextExpectedANumberParam	int <b>Value:30</b>
Constants.FP_HypertextValidateErr	int <b>Value:2304</b>
Constants.FV_HypertextValid	int <b>Value:0</b>
Constants.FV_HypertextUsesDefaultText	int <b>Value:200</b>
Constants.FV_HypertextFileNotRegular	int <b>Value:210</b>
Constants.FV_HypertextFileNotMakerDoc	int <b>Value:211</b>
Constants.FV_HypertextCantOpenDestFile	int <b>Value:212</b>
Constants.FV_HypertextDestinationLinkNotFound	int <b>Value:220</b>
Constants.FV_HypertextDuplicateLinkName	int <b>Value:221</b>
Constants.FV_HypertextPageNameNotFound	int <b>Value:230</b>
Constants.FV_HypertextUnrecognizedObjectType	int <b>Value:240</b>
Constants.FV_HypertextObjectIDNotFound	int <b>Value:241</b>

Property name	Data Type
Constants.FV_HypertextBadMatrixSize	int <b>Value:250</b>
Constants.FV_HypertextMatrixCommandInvalid	int <b>Value:251</b>
Constants.FV_HypertextFlowMissingLines	int <b>Value:252</b>
Constants.FV_HypertextNoNamedFlow	int <b>Value:260</b>
Constants.FV_HypertextRecursiveFlow	int <b>Value:261</b>
Constants.FV_HypertextMissingPopupMarker	int <b>Value:270</b>
Constants.FV_HypertextMissingPopupLabelItem	int <b>Value:271</b>
Constants.FV_HypertextEmptyLineInMiddleOfPopup	int <b>Value:272</b>
Constants.FV_HypertextCommandIllegalWithinPopup	int <b>Value:273</b>
Constants.FV_HypertextFcodeInvalid	int <b>Value:280</b>
Constants.FP_HypertextParseBadParam	int <b>Value:2305</b>
Constants.FP_HypertextParseErrMsg	int <b>Value:2306</b>
Constants.FP_HypertextParsedCmdCode	int <b>Value:2307</b>
Constants.FV_CmdNotTyped	int <b>Value:0</b>
Constants.FV_CmdError	int <b>Value:1</b>
Constants.FV_CmdUnknown	int <b>Value:2</b>
Constants.FV_CmdNoop	int <b>Value:3</b>

Property name	Data Type
Constants.FV_CmdAlert	int <b>Value:8</b>
Constants.FV_CmdAlertTitle	int <b>Value:9</b>
Constants.FV_CmdExit	int <b>Value:10</b>
Constants.FV_CmdGoToLink	int <b>Value:11</b>
Constants.FV_CmdGoToLinkFitWin	int <b>Value:12</b>
Constants.FV_CmdGoToNew	int <b>Value:13</b>
Constants.FV_CmdGoToPage	int <b>Value:14</b>
Constants.FV_CmdGoToObjectId	int <b>Value:15</b>
Constants.FV_CmdGoToObjectIdFitWin	int <b>Value:16</b>
Constants.FV_CmdMatrix	int <b>Value:17</b>
Constants.FV_CmdMessage	int <b>Value:18</b>
Constants.FV_CmdNewLink	int <b>Value:19</b>
Constants.FV_CmdNextPage	int <b>Value:20</b>
Constants.FV_CmdPreviousPage	int <b>Value:21</b>
Constants.FV_CmdOpenLink	int <b>Value:22</b>
Constants.FV_CmdOpenLinkFitWin	int <b>Value:23</b>
Constants.FV_CmdOpenNew	int <b>Value:24</b>

Property name	Data Type
Constants.FV_CmdOpenObjectId	int <b>Value:25</b>
Constants.FV_CmdOpenObjectIdFitWin	int <b>Value:26</b>
Constants.FV_CmdOpenPage	int <b>Value:27</b>
Constants.FV_CmdPopup	int <b>Value:28</b>
Constants.FV_CmdPreviousLink	int <b>Value:29</b>
Constants.FV_CmdPreviousLinkFitWin	int <b>Value:30</b>
Constants.FV_CmdQuit	int <b>Value:31</b>
Constants.FV_CmdQuitAll	int <b>Value:32</b>
Constants.FV_CmdMultiMedia	int <b>Value:33</b>
Constants.FV_CmdFCodes	int <b>Value:256</b>
Constants.FV_CmdOutCodes	int <b>Value:257</b>
Constants.FV_CmdInCodes	int <b>Value:258</b>
Constants.FV_CmdApplyMathRules	int <b>Value:259</b>
Constants.FV_CmdThesaurusLookup	int <b>Value:260</b>
Constants.FV_CmdNative	int <b>Value:261</b>
Constants.FV_CmdHelpLink	int <b>Value:262</b>
Constants.FV_CmdBeginRange	int <b>Value:263</b>

Property name	Data Type
Constants.FV_CmdEndRange	int <b>Value:264</b>
Constants.FP_HypertextParsedCmdDest	int <b>Value:2308</b>
Constants.FV_DestNowhere	int <b>Value:0</b>
Constants.FV_DestMarkerNewLink	int <b>Value:1</b>
Constants.FV_DestFirstPage	int <b>Value:2</b>
Constants.FV_DestLastPage	int <b>Value:3</b>
Constants.FV_DestPageNum	int <b>Value:4</b>
Constants.FV_DestFluidFlow	int <b>Value:5</b>
Constants.FV_DestMarker	int <b>Value:6</b>
Constants.FV_DestObjectId	int <b>Value:7</b>
Constants.FV_DestXRef	int <b>Value:8</b>
Constants.FP_HypertextParsedCmdDest ObjType	int <b>Value:2309</b>
Constants.FV_ObjectUnknown	int <b>Value:0</b>
Constants.FV_ObjectMarker	int <b>Value:1</b>
Constants.FV_ObjectPgf	int <b>Value:2</b>
Constants.FV_ObjectXref	int <b>Value:3</b>
Constants.FV_ObjectGraphic	int <b>Value:4</b>

Property name	Data Type
Constants.FV_ObjectElement	int <b>Value:5</b>
Constants.FV_ObjectTextInset	int <b>Value:6</b>
Constants.FV_ObjectDataLink	int <b>Value:7</b>
Constants.FP_HypertextParsedCmdDestObjID	int <b>Value:2310</b>
Constants.FP_HypertextParsedCmdMatrixRows	int <b>Value:2311</b>
Constants.FP_HypertextParsedCmdMatrixColumns	int <b>Value:2312</b>
Constants.FP_HypertextParsedLinkName	int <b>Value:2313</b>
Constants.FP_HypertextParsedPageName	int <b>Value:2314</b>
Constants.FP_HypertextParsedFlowName	int <b>Value:2315</b>
Constants.FP_HypertextParsedRangeName	int <b>Value:2316</b>
Constants.FP_HypertextParsedClientName	int <b>Value:2317</b>
Constants.FP_HypertextParsedTitle	int <b>Value:2318</b>
Constants.FP_HypertextParsedMessage	int <b>Value:2319</b>
Constants.FP_HypertextParsedDIFilename	int <b>Value:2320</b>
Constants.FP_FirstElementDefInDoc	int <b>Value:1483</b>
Constants.FP_ElementBoundaryDisplay	int <b>Value:1484</b>
Constants.FP_BannerTextDisplay	int <b>Value:1600</b>



Property name	Data Type
Constants.FP_BooleanConditionExpression	int <b>Value:</b> 2321
Constants.FP_BooleanConditionState	int <b>Value:</b> 2322
Constants.FP_BooleanConditionExpressionTag	int <b>Value:</b> 995
Constants.FP_IsDitapInResourceManager	int <b>Value:</b> 2332
Constants.FP_IsDocDitap	int <b>Value:</b> 2879
Constants.FP_IsDocDita	int <b>Value:</b> 2880
Constants.FP_DirectOLESupportInXml	int <b>Value:</b> 2881
Constants.FP_DocIsStructuredDummy	int <b>Value:</b> 2882
Constants.FP_StructuredDummyDocText	int <b>Value:</b> 2883
Constants.FV_ELEM_DISP_NONE	int <b>Value:</b> 0
Constants.FV_ELEM_DISP_BRACKETS	int <b>Value:</b> 1
Constants.FV_ELEM_DISP_TAGS	int <b>Value:</b> 2
Constants.FP_ElementCatalogDisplay	int <b>Value:</b> 1485
Constants.FV_ELCAT_STRICT	int <b>Value:</b> 0x00
Constants.FV_ELCAT_LOOSE	int <b>Value:</b> 0x01
Constants.FV_ELCAT_CHILDREN	int <b>Value:</b> 0x02
Constants.FV_ELCAT_ALL	int <b>Value:</b> 0x03

Property name	Data Type
Constants.FV_ELCAT_CUSTOM	int <b>Value:</b> 0x04
Constants.FP_DefaultInclusions	int <b>Value:</b> 1486
Constants.FP_DefaultExclusions	int <b>Value:</b> 1487
Constants.FP_ElementCatalog	int <b>Value:</b> 1488
Constants.FP_FirstFmtChangeListInDoc	int <b>Value:</b> 1489
Constants.FP_NewElemAttrDisplay	int <b>Value:</b> 1490
Constants.FV_ATTR_DISP_NONE	int <b>Value:</b> 1
Constants.FV_ATTR_DISP_REQSPEC	int <b>Value:</b> 2
Constants.FV_ATTR_DISP_ALL	int <b>Value:</b> 3
Constants.FP_NewElemAttrEditing	int <b>Value:</b> 1491
Constants.FV_ATTR_EDIT_NONE	int <b>Value:</b> 0
Constants.FV_ATTR_EDIT_REQUIRED	int <b>Value:</b> 1
Constants.FV_ATTR_EDIT_ALWAYS	int <b>Value:</b> 2
Constants.FP_UseInitialStructure	int <b>Value:</b> 1492
Constants.FP_SeparateInclusions	int <b>Value:</b> 1493
Constants.FP_ElementSelection	int <b>Value:</b> 1494
Constants.FP_StructuredApplication	int <b>Value:</b> 1495

Property name	Data Type
Constants.FP_CustomElementList	int <b>Value:</b> 1496
Constants.FP_StructuredApplicationList	int <b>Value:</b> 1497
Constants.FP_StructuredApplicationForOpen	int <b>Value:</b> 1498
Constants.FP_UseInitialStructureOfAutoInsertedElements	int <b>Value:</b> 1499
Constants.FP_StructAppAttrConfigFile	int <b>Value:</b> 1500
Constants.FP_ShowElementDescriptiveNames	int <b>Value:</b> 1501
Constants.FP_BkGndColor	int <b>Value:</b> 1502
Constants.FP_TextColor	int <b>Value:</b> 1503
Constants.FP_DitaMode	int <b>Value:</b> 1504
Constants.FP_XSLTTransformationScenarioFile	int <b>Value:</b> 1505
Constants.FP_XSLTProcessors	int <b>Value:</b> 1506
Constants.FV_STRICTLY_VALID	int <b>Value:</b> 0x01
Constants.FV_LOOSELY_VALID	int <b>Value:</b> 0x02
Constants.FV_ALTERNATIVE	int <b>Value:</b> 0x04
Constants.FV_INCLUSION	int <b>Value:</b> 0x08
Constants.FP_HighestLevelElement	int <b>Value:</b> 1509
Constants.FP_ComponentElement	int <b>Value:</b> 1551

Property name	Data Type
Constants.FP_ExtractElementTags	int <b>Value:1552</b>
Constants.FP_NextElementDefInDoc	int <b>Value:1572</b>
Constants.FP_ElementInCatalog	int <b>Value:1573</b>
Constants.FP_ValidHighestLevel	int <b>Value:1574</b>
Constants.FP_GeneralRule	int <b>Value:1575</b>
Constants.FP_Exclusions	int <b>Value:1576</b>
Constants.FP_Inclusions	int <b>Value:1577</b>
Constants.FP_GeneralRuleErrorOffsets	int <b>Value:1578</b>
Constants.FP_ElementDefType	int <b>Value:1579</b>
Constants.FP_Comment	int <b>Value:1581</b>
Constants.FP_TextFmtRules	int <b>Value:1582</b>
Constants.FP_ObjectFmtRules	int <b>Value:1583</b>
Constants.FP_AttributeDefs	int <b>Value:1584</b>
Constants.FP_InitStructurePattern	int <b>Value:1585</b>
Constants.FP_TableTagging	int <b>Value:1585</b>
Constants.FP_ElementPgfFormat	int <b>Value:1586</b>
Constants.FP_PrefixRules	int <b>Value:1587</b>

Property name	Data Type
Constants.FP_SuffixRules	int <b>Value:1588</b>
Constants.FP_FirstPgfrules	int <b>Value:1589</b>
Constants.FP_LastPgfrules	int <b>Value:1590</b>
Constants.FP_AlsoInsert	int <b>Value:1591</b>
Constants.FP_ParsedGeneralRule	int <b>Value:1592</b>
Constants.FP_ElementDataType	int <b>Value:1593</b>
Constants.FP_AllowedDataRange	int <b>Value:1594</b>
Constants.FP_AlsoInserts	int <b>Value:1595</b>
Constants.FP_DescriptiveTag	int <b>Value:1596</b>
Constants.FP_ElementDescription	int <b>Value:1597</b>
Constants.FP_XMLStructuredApplication List	int <b>Value:1598</b>
Constants.FP_BannerText	int <b>Value:1599</b>
Constants.FV_FO_UNSPECIFIED	int <b>Value:0</b>
Constants.FV_FO_TEXTNODE	int <b>Value:0</b>
Constants.FV_FO_CONTAINER	int <b>Value:1</b>
Constants.FV_FO_SYS_VAR	int <b>Value:2</b>
Constants.FV_FO_XREF	int <b>Value:3</b>

Property name	Data Type
Constants.FV_FO_MARKER	int <b>Value:4</b>
Constants.FV_FO_FOOTNOTE	int <b>Value:5</b>
Constants.FV_FO_GRAPHIC	int <b>Value:6</b>
Constants.FV_FO_EQN	int <b>Value:7</b>
Constants.FV_FO_TBL	int <b>Value:8</b>
Constants.FV_FO_TBL_TITLE	int <b>Value:9</b>
Constants.FV_FO_TBL_HEADING	int <b>Value:10</b>
Constants.FV_FO_TBL_BODY	int <b>Value:11</b>
Constants.FV_FO_TBL_FOOTING	int <b>Value:12</b>
Constants.FV_FO_TBL_ROW	int <b>Value:13</b>
Constants.FV_FO_TBL_CELL	int <b>Value:14</b>
Constants.FV_FO_RUBI_GROUP	int <b>Value:15</b>
Constants.FV_FO_RUBI	int <b>Value:16</b>
Constants.FV_FO_NUMTYPES	int <b>Value:17</b>
Constants.FV_DT_UNSPECIFIED	int <b>Value:0</b>
Constants.FV_DT_INTEGER	int <b>Value:1</b>
Constants.FV_DT_FLOAT	int <b>Value:2</b>

Property name	Data Type
Constants.FV_DT_NUMTYPES	int <b>Value:3</b>
Constants.FV_AT_STRING	int <b>Value:0</b>
Constants.FV_AT_STRINGS	int <b>Value:1</b>
Constants.FV_AT_CHOICES	int <b>Value:2</b>
Constants.FV_AT_INTEGER	int <b>Value:3</b>
Constants.FV_AT_INTEGERS	int <b>Value:4</b>
Constants.FV_AT_REAL	int <b>Value:5</b>
Constants.FV_AT_REALS	int <b>Value:6</b>
Constants.FV_AT_UNIQUE_ID	int <b>Value:7</b>
Constants.FV_AT_UNIQUE_IDREF	int <b>Value:8</b>
Constants.FV_AT_UNIQUE_IDREFS	int <b>Value:9</b>
Constants.FV_AT_NUMTYPES	int <b>Value:10</b>
Constants.FV_AF_READ_ONLY	int <b>Value:0x0001</b>
Constants.FV_AF_HIDDEN	int <b>Value:0x0002</b>
Constants.FV_AF_FIXED	int <b>Value:0x0004</b>
Constants.FP_FmtRuleType	int <b>Value:2100</b>
Constants.FP_CountElements	int <b>Value:2101</b>

Property name	Data Type
Constants.FP_StopCountingAt	int <b>Value:</b> 2102
Constants.FP_FmtRuleClauses	int <b>Value:</b> 2103
Constants.FP_FmtRuleClause	int <b>Value:</b> 2104
Constants.FV_CONTEXT_RULE	int <b>Value:</b> 0
Constants.FV_LEVEL_RULE	int <b>Value:</b> 1
Constants.FP_Specification	int <b>Value:</b> 2150
Constants.FP_IsTextRange	int <b>Value:</b> 2151
Constants.FP_RuleClauseType	int <b>Value:</b> 2152
Constants.FP_FormatTag	int <b>Value:</b> 2153
Constants.FP_FmtChangeListTag	int <b>Value:</b> 2154
Constants.FP_FmtChangeList	int <b>Value:</b> 2155
Constants.FP_SubFmtRule	int <b>Value:</b> 2156
Constants.FP_ContextLabel	int <b>Value:</b> 2157
Constants.FP_ElemPrefixSuffix	int <b>Value:</b> 2158
Constants.FP_FmtRule	int <b>Value:</b> 2159
Constants.FP_ParsedSpecification	int <b>Value:</b> 2160
Constants.FP_SpecificationForCSS	int <b>Value:</b> 2161



Property name	Data Type
Constants.FP_ContextForCSS	int <b>Value:2162</b>
Constants.FP_ParseFullSpecification	int <b>Value:2163</b>
Constants.FV_RC_TAG	int <b>Value:0</b>
Constants.FV_RC_SUB_FMTRULE	int <b>Value:1</b>
Constants.FV_RC_CHANGELIST	int <b>Value:2</b>
Constants.FV_RC_CHANGELIST_TAG	int <b>Value:3</b>
Constants.FV_EDT_TEXT	int <b>Value:40</b>
Constants.FV_EDT_TEXTONLY	int <b>Value:41</b>
Constants.FV_EDT_ANY	int <b>Value:42</b>
Constants.FV_EDT_ALL	int <b>Value:43</b>
Constants.FV_EDT_FIRST	int <b>Value:44</b>
Constants.FV_EDT_MIDDLE	int <b>Value:45</b>
Constants.FV_EDT_LAST	int <b>Value:46</b>
Constants.FV_EDT_NOTFIRST	int <b>Value:47</b>
Constants.FV_EDT_NOTLAST	int <b>Value:48</b>
Constants.FV_EDT_AFTER	int <b>Value:49</b>
Constants.FV_EDT_BEFORE	int <b>Value:50</b>

Property name	Data Type
Constants.FV_EDT_BETWEEN	int <b>Value:53</b>
Constants.FV_EDT_INSERTPAR	int <b>Value:23</b>
Constants.FV_EDT_ONLY	int <b>Value:62</b>
Constants.FV_EDT_ATTRGRPO	int <b>Value:30</b>
Constants.FV_EDT_ELEMENT	int <b>Value:25</b>
Constants.FP_ElementDef	int <b>Value:1621</b>
Constants.FP_ParentElement	int <b>Value:1622</b>
Constants.FP_FirstChildElement	int <b>Value:1623</b>
Constants.FP_LastChildElement	int <b>Value:1624</b>
Constants.FP_PrevSiblingElement	int <b>Value:1625</b>
Constants.FP_NextSiblingElement	int <b>Value:1626</b>
Constants.FP_ElementIsCollapsed	int <b>Value:1627</b>
Constants.FP_BookComponent	int <b>Value:1628</b>
Constants.FP_Object	int <b>Value:1630</b>
Constants.FP_MatchingTextClauses	int <b>Value:1631</b>
Constants.FP_MatchingObjectClauses	int <b>Value:1632</b>
Constants.FP_Attributes	int <b>Value:1634</b>

Property name	Data Type
Constants.FP_AttributesEx	int <b>Value:</b> 1670
Constants.FP_AttrDisplay	int <b>Value:</b> 1635
Constants.FP_ElementType	int <b>Value:</b> 1636
Constants.FP_IDAttrValue	int <b>Value:</b> 1637
Constants.FP_MatchingFirstPgfClauses	int <b>Value:</b> 1638
Constants.FP_MatchingLastPgfClauses	int <b>Value:</b> 1639
Constants.FP_MatchingPrefixClauses	int <b>Value:</b> 1640
Constants.FP_MatchingSuffixClauses	int <b>Value:</b> 1641
Constants.FP_MatchesContextInUserString	int <b>Value:</b> 1642
Constants.FP_Namespace	int <b>Value:</b> 1643
Constants.FP_NamespaceScope	int <b>Value:</b> 1644
Constants.FP_NumNamespaces	int <b>Value:</b> 1645
Constants.FP_NextElementDFS	int <b>Value:</b> 2334
Constants.FP_PrevElementDFS	int <b>Value:</b> 2335
Constants.FP_ElementIsTopicRef	int <b>Value:</b> 2336
Constants.FP_ElementIsTopicHead	int <b>Value:</b> 2337
Constants.FP_ElementIsTopicGroup	int <b>Value:</b> 2338

Property name	Data Type
Constants.FP_TopicRefAbsolutePath	int <b>Value:</b> 2347
Constants.FP_Element	int <b>Value:</b> 2170
Constants.FP_TblElement	int <b>Value:</b> 2171
Constants.FP_TblTitleElement	int <b>Value:</b> 2172
Constants.FP_TblHeaderElement	int <b>Value:</b> 2173
Constants.FP_TblBodyElement	int <b>Value:</b> 2174
Constants.FP_TblFooterElement	int <b>Value:</b> 2175
Constants.FP_RubiElement	int <b>Value:</b> 2176
Constants.FP_AllowAsSpecialCase	int <b>Value:</b> 1649
Constants.FP_ElementIsUndefined	int <b>Value:</b> 1650
Constants.FP_ContentIsStrictlyValid	int <b>Value:</b> 1651
Constants.FP_ContentIsLooselyValid	int <b>Value:</b> 1652
Constants.FP_ContentNeededAtBegin	int <b>Value:</b> 1653
Constants.FP_HoleAtEnd	int <b>Value:</b> 1654
Constants.FP_ContentNeededAtEnd	int <b>Value:</b> 1654
Constants.FP_ElementIsExcludedInContext	int <b>Value:</b> 1655
Constants.FP_ElementIsInvalidInParent	int <b>Value:</b> 1656

Property name	Data Type
Constants.FP_ElementIsInvalidInPosition	int <b>Value:</b> 1657
Constants.FP_ElementTypeMismatch	int <b>Value:</b> 1658
Constants.FP_HoleBeforeElement	int <b>Value:</b> 1659
Constants.FP_TextIsInvalidInElement	int <b>Value:</b> 1660
Constants.FP_InvalidHighestLevel	int <b>Value:</b> 1661
Constants.FP_BookComponentMissing	int <b>Value:</b> 1662
Constants.FP_ErrorInBookComponent	int <b>Value:</b> 1663
Constants.FP_NextInvalidElement	int <b>Value:</b> 1664
Constants.FP_ContentMustBeEmpty	int <b>Value:</b> 1665
Constants.FP_AttributeValueRequired	int <b>Value:</b> 1666
Constants.FP_AttributeValueInvalid	int <b>Value:</b> 1667
Constants.FP_ValidationFlags	int <b>Value:</b> 1668
Constants.FP_ElementMarkedForNamed Destination	int <b>Value:</b> 1669
Constants.FV_ELEM_UNDEFINED	int <b>Value:</b> 0x0001
Constants.FV_ELEM_TYPE_MISMATCH	int <b>Value:</b> 0x0002
Constants.FV_ELEM_EXCLUDED	int <b>Value:</b> 0x0004
Constants.FV_ELEM_INVALID_IN_PARENT	int <b>Value:</b> 0x0008

Property name	Data Type
Constants.FV_ELEM_INVALID_AT_POSITION	int <b>Value:</b> 0x0010
Constants.FV_ELEM_HAS_TEXT_INVALID	int <b>Value:</b> 0x0020
Constants.FV_ELEM_CONTENT_MUST_BE_EMPTY	int <b>Value:</b> 0x0040
Constants.FV_ELEM_MISSING_CONTENT_BEFORE	int <b>Value:</b> 0x0080
Constants.FV_ELEM_MISSING_CONTENT_AT_BEG	int <b>Value:</b> 0x0100
Constants.FV_ELEM_MISSING_CONTENT_AT_END	int <b>Value:</b> 0x0200
Constants.FV_ELEM_NOT_VALID_AS_ROOT	int <b>Value:</b> 0x0400
Constants.FV_ELEM_BOOK_COMP_MISSING	int <b>Value:</b> 0x0800
Constants.FV_ELEM_BOOK_COMP_INVALID	int <b>Value:</b> 0x1000
Constants.FV_ELEM_ATTRVAL_REQUIRED	int <b>Value:</b> 0x2000
Constants.FV_ELEM_ATTRVAL_INVALID	int <b>Value:</b> 0x4000
Constants.FV_ELEM_CONTENT_STRICTLY_VALID	int <b>Value:</b> 0x10000
Constants.FV_ELEM_CONTENT_LOOSELY_VALID	int <b>Value:</b> 0x20000
Constants.FV_ELEM_INVISIBLE_ATTRVAL_REQUIRED	int <b>Value:</b> 0x40000
Constants.FV_ELEM_INVISIBLE_ATTRVAL_INVALID	int <b>Value:</b> 0x80000
Constants.FV_AV_REQUIRED	int <b>Value:</b> 1
Constants.FV_AV_INVALID_CHOICE	int <b>Value:</b> 2

Property name	Data Type
Constants.FV_AV_INVALID_FORMAT	int <b>Value:3</b>
Constants.FV_AV_IDREF_UNRESOLVED	int <b>Value:4</b>
Constants.FV_AV_ID_DUPLICATE_IN_DOC	int <b>Value:5</b>
Constants.FV_AV_ID_DUPLICATE_IN_BOOK	int <b>Value:6</b>
Constants.FV_AV_TOO_MANY_TOKENS	int <b>Value:7</b>
Constants.FV_AV_UNDEFINED	int <b>Value:8</b>
Constants.FV_AV_OUT_OF_RANGE	int <b>Value:9</b>
Constants.FV_OVERRIDDEN_DEFAULT	int <b>Value:0x00</b>
Constants.FV_OVERRIDDEN_NOVALUE	int <b>Value:0x01</b>
Constants.FV_OVERRIDDEN_CONREF_DUCT	int <b>Value:0x02</b>
Constants.FV_OVERRIDDEN_KEYDEF_TO_KEYREF	int <b>Value:0x03</b>
Constants.FV_Element	int <b>Value:0</b>
Constants.FV_ElementsOfType	int <b>Value:1</b>
Constants.FV_AllElements	int <b>Value:2</b>
Constants.FP_NextFmtChangeListInDoc	int <b>Value:2000</b>
Constants.FP_FmtChangeListInCatalog	int <b>Value:2021</b>
Constants.FP_PgfCatalogReference	int <b>Value:2001</b>

Property name	Data Type
Constants.FP_SpaceAboveChange	int <b>Value:</b> 2002
Constants.FP_SpaceBelowChange	int <b>Value:</b> 2003
Constants.FP_LeftIndentChange	int <b>Value:</b> 2004
Constants.FP_FirstIndentChange	int <b>Value:</b> 2005
Constants.FP_RightIndentChange	int <b>Value:</b> 2006
Constants.FP_LeadingChange	int <b>Value:</b> 2007
Constants.FP_LineSpacingFixed	int <b>Value:</b> 2008
Constants.FP_TopSepAtIndent	int <b>Value:</b> 2009
Constants.FP_BottomSepAtIndent	int <b>Value:</b> 2010
Constants.FP_FontSizeChange	int <b>Value:</b> 2012
Constants.FP_SpreadChange	int <b>Value:</b> 2013
Constants.FP_MoveTabs	int <b>Value:</b> 2014
Constants.FP_CellTopMarginFixed	int <b>Value:</b> 2015
Constants.FP_CellBottomMarginFixed	int <b>Value:</b> 2016
Constants.FP_CellLeftMarginFixed	int <b>Value:</b> 2017
Constants.FP_CellRightMarginFixed	int <b>Value:</b> 2018
Constants.FP_FirstIndentIsRelative	int <b>Value:</b> 2019



Property name	Data Type
Constants.FP_FirstIndentRelPos	int <b>Value:</b> 2020
Constants.FP_CellTopMarginChange	int <b>Value:</b> 2022
Constants.FP_CellBottomMarginChange	int <b>Value:</b> 2023
Constants.FP_CellLeftMarginChange	int <b>Value:</b> 2024
Constants.FP_CellRightMarginChange	int <b>Value:</b> 2025
Constants.FP_StretchChange	int <b>Value:</b> 2026
Constants.FF_IED_REMOVE_OVERRIDES	int <b>Value:</b> 0x0001
Constants.FF_IED_REMOVE_BOOK_INFO	int <b>Value:</b> 0x0002
Constants.FF_IED_DO_NOT_IMPORT_ED D	int <b>Value:</b> 0x0004
Constants.FF_IED_NO_NOTIFY	int <b>Value:</b> 0x0008
Constants.FF_IED_DELETE_EMPTY_PAG ES	int <b>Value:</b> 0x0010
Constants.FF_IED_DO_NOT_IMPORT_ST RUCTAPP	int <b>Value:</b> 0x0040
Constants.FP_Text	int <b>Value:</b> 1701
Constants.FP_State	int <b>Value:</b> 1702
Constants.FV_DlgOptNotActive	int <b>Value:</b> 0
Constants.FV_DlgOptActive	int <b>Value:</b> 1
Constants.FV_DlgOptDontCare	int <b>Value:</b> 2

Property name	Data Type
Constants.FP_Labels	int <b>Value:1703</b>
Constants.FP_NumLines	int <b>Value:1704</b>
Constants.FP_FirstVis	int <b>Value:1705</b>
Constants.FP_DoubleClick	int <b>Value:1706</b>
Constants.FP_NumItems	int <b>Value:1707</b>
Constants.FP_Sensitivity	int <b>Value:1708</b>
Constants.FP_MinVal	int <b>Value:1709</b>
Constants.FP_MaxVal	int <b>Value:1710</b>
Constants.FP_IncrVal	int <b>Value:1711</b>
Constants.FP_Size	int <b>Value:1712</b>
Constants.FP_Visibility	int <b>Value:1713</b>
Constants.FP_PasswordStyle	int <b>Value:1714</b>
Constants.FP_ReadOnly	int <b>Value:1723</b>
Constants.FP_TabStops	int <b>Value:1715</b>
Constants.FP_Icon	int <b>Value:1716</b>
Constants.FP_WindowHandle	int <b>Value:1717</b>
Constants.FP_SortingEnabled	int <b>Value:1718</b>

Property name	Data Type
Constants.FP_SelectedRow	int <b>Value:1730</b>
Constants.FP_SelectedRowId	int <b>Value:1731</b>
Constants.FP_SelectedRowItems	int <b>Value:1719</b>
Constants.FP_LabelColor	int <b>Value:1720</b>
Constants.FP_LabelBold	int <b>Value:1721</b>
Constants.FP_Box_BG_Color	int <b>Value:1722</b>
Constants.FV_DlgClose	int <b>Value:-1</b>
Constants.FV_DlgReset	int <b>Value:-2</b>
Constants.FV_DlgNoChange	int <b>Value:-3</b>
Constants.FV_DlgEnter	int <b>Value:-4</b>
Constants.FV_DlgUndo	int <b>Value:-5</b>
Constants.FV_DlgPrevPage	int <b>Value:-6</b>
Constants.FV_DlgNextPage	int <b>Value:-7</b>
Constants.FV_DlgNeedsUpdate	int <b>Value:-8</b>
Constants.FV_DlgInit	int <b>Value:-9</b>
Constants.FV_DlgHide	int <b>Value:-10</b>
Constants.FV_DlgShow	int <b>Value:-11</b>

Property name	Data Type
Constants.FV_DlgResize	int <b>Value:-12</b>
Constants.FV_DlgPrefSetInitialState	int <b>Value:-100</b>
Constants.FV_DlgPrefVerify	int <b>Value:-101</b>
Constants.FV_DlgPrefCommit	int <b>Value:-102</b>
Constants.FV_EvShift	int <b>Value:0x0001</b>
Constants.FV_EvControl	int <b>Value:0x0002</b>
Constants.FV_EvMeta	int <b>Value:0x0004</b>
Constants.FV_EvOption	int <b>Value:0x0004</b>
Constants.FV_EvCaps	int <b>Value:0x0008</b>
Constants.FV_EvCommand	int <b>Value:0x0010</b>
Constants.FV_EvMax	int <b>Value:0x0010</b>
Constants.FP_DbItemNum	int <b>Value:1800</b>
Constants.FP_DbType	int <b>Value:1801</b>
Constants.FP_DbIdentifier	int <b>Value:1802</b>
Constants.FP_DbVarLabelWidth	int <b>Value:1803</b>
Constants.FP_DbStuffItem	int <b>Value:1804</b>
Constants.FP_DbSbxNumLines	int <b>Value:1805</b>

Property name	Data Type
Constants.FP_DbRadioButtonGroup	int <b>Value:1806</b>
Constants.FP_DbCheckBoxState	int <b>Value:1807</b>
Constants.FP_DbFirstFocus	int <b>Value:1808</b>
Constants.FP_DbDefaultButton	int <b>Value:1809</b>
Constants.FP_DbOKButton	int <b>Value:1810</b>
Constants.FP_DbCancelButton	int <b>Value:1811</b>
Constants.FP_DbFbTextBox	int <b>Value:1812</b>
Constants.FP_DbFbScrollBar	int <b>Value:1813</b>
Constants.FP_DbFbCurrentDir	int <b>Value:1814</b>
Constants.FP_DbFbStatus	int <b>Value:1815</b>
Constants.FP_DbtitleLabel	int <b>Value:1816</b>
Constants.FP_DbAttributes	int <b>Value:1817</b>
Constants.FP_DbEditable	int <b>Value:1818</b>
Constants.FP_DbNoHelp	int <b>Value:1819</b>
Constants.FP_LineAscent	int <b>Value:1900</b>
Constants.FP_LineDescent	int <b>Value:1901</b>
Constants.FP_LineBaseline	int <b>Value:1902</b>

Property name	Data Type
Constants.FP_MenuItemIsEnabled	int <b>Value:1922</b>
Constants.FP_NextMenuItemInMenu	int <b>Value:1923</b>
Constants.FP_PrevMenuItemInMenu	int <b>Value:1924</b>
Constants.FP_NextMenuItemInSession	int <b>Value:1925</b>
Constants.FP_MenuType	int <b>Value:1926</b>
Constants.FV_MENU_MENUBAR	int <b>Value:1</b>
Constants.FV_MENU_POPUP	int <b>Value:2</b>
Constants.FV_MENU_ADHOCRULER	int <b>Value:3</b>
Constants.FV_MENU_DEFAULT	int <b>Value:4</b>
Constants.FP_FirstMenuItemInMenu	int <b>Value:1927</b>
Constants.FP_CommandNum	int <b>Value:1928</b>
Constants.FP_KeyboardShortcutLabel	int <b>Value:1929</b>
Constants.FP_KeyboardShortcuts	int <b>Value:1930</b>
Constants.FP_CanHaveCheckMark	int <b>Value:1931</b>
Constants.FP_CheckMarkIsOn	int <b>Value:1932</b>
Constants.FP_MenuItemType	int <b>Value:1933</b>
Constants.FV_MENUITEM_FRAME	int <b>Value:1</b>

Property name	Data Type
Constants.FV_MENUITEM_API	int <b>Value:2</b>
Constants.FV_MENUITEM_MACRO	int <b>Value:3</b>
Constants.FV_MENUITEM_EXPANDOMATIC	int <b>Value:4</b>
Constants.FP_ExpandOMaticParent	int <b>Value:1934</b>
Constants.FP_EnabledWhen	int <b>Value:1935</b>
Constants.FV_ENABLE_ALWAYS_ENABLE	int <b>Value:1</b>
Constants.FV_ENABLE_ALWAYS_DISABLE	int <b>Value:2</b>
Constants.FV_ENABLE_IN_PARA_TEXT	int <b>Value:3</b>
Constants.FV_ENABLE_IN_TEXT_LINE	int <b>Value:4</b>
Constants.FV_ENABLE_IS_TEXT_SEL	int <b>Value:5</b>
Constants.FV_ENABLE_IN_MATH	int <b>Value:6</b>
Constants.FV_ENABLE_IN_TEXT	int <b>Value:7</b>
Constants.FV_ENABLE_OBJ_PROPS	int <b>Value:8</b>
Constants.FV_ENABLE_IN_TABLE	int <b>Value:9</b>
Constants.FV_ENABLE_IN_TABLE_TITLE	int <b>Value:10</b>
Constants.FV_ENABLE_IN_CELL_TEXT	int <b>Value:11</b>
Constants.FV_ENABLE_IS_CELL	int <b>Value:12</b>

Property name	Data Type
Constants.FV_ENABLE_IS_CELLS	int <b>Value:13</b>
Constants.FV_ENABLE_IS_TABLE	int <b>Value:14</b>
Constants.FV_ENABLE_IS_OBJ	int <b>Value:15</b>
Constants.FV_ENABLE_IS_TEXT_FRAME	int <b>Value:16</b>
Constants.FV_ENABLE_IS_OR_IN_FRAME	int <b>Value:17</b>
Constants.FV_ENABLE_IS_AFRAME	int <b>Value:18</b>
Constants.FV_ENABLE_IS_TEXT_INSET	int <b>Value:19</b>
Constants.FV_ENABLE_IS_GRAPHIC_INSET	int <b>Value:20</b>
Constants.FV_ENABLE_IN_FLOW	int <b>Value:21</b>
Constants.FV_ENABLE_COPY	int <b>Value:22</b>
Constants.FV_ENABLE_COPY_FONT	int <b>Value:23</b>
Constants.FV_ENABLE_CAN_PASTE	int <b>Value:24</b>
Constants.FV_ENABLE_IS_VIEW_ONLY	int <b>Value:25</b>
Constants.FV_ENABLE_NEEDS_DOCP_ONLY	int <b>Value:26</b>
Constants.FV_ENABLE_NEEDS_BOOKP_ONLY	int <b>Value:27</b>
Constants.FV_ENABLE_NEEDS_DOCP_OR_BOOKP	int <b>Value:28</b>
Constants.FV_ENABLE_BOOK_HAS_SELECTION	int <b>Value:29</b>



Property name	Data Type
Constants.FV_ENABLE_DOC_OR_BOOK_HAS_SELECTION	int Value:30
Constants.FP_Fcode	int Value:1936
Constants.FP_Fcodes	int Value:1937
Constants.FP_HelpLink	int Value:1938
Constants.FP_HasShiftOrUnshiftCommand	int Value:1939
Constants.FV_ITEM_HAS_SHIFT_COMMAND	int Value:1
Constants.FV_ITEM_HAS_UNSHIFT_COMMAND	int Value:2
Constants.FV_ITEM_HAS_NO_SHIFT_OR_UNSHIFT_COMMAND	int Value:3
Constants.FP_ShiftOrUnshiftCommand	int Value:1940
Constants.FP_Mode	int Value:1941
Constants.FV_MODE_MATH	int Value:1
Constants.FV_MODE_NONMATH	int Value:2
Constants.FV_MODE_ALL	int Value:3
Constants.FP_NextCommandInSession	int Value:1942
Constants.FP_Hypertext	int Value:1943
Constants.FP_NextTiInDoc	int Value:2050
Constants.FP_TiAutomaticUpdate	int Value:2051

Property name	Data Type
Constants.FV_UpdateAllClientTi	int <b>Value:1</b>
Constants.FV_UpdateAllAutomaticClientTi	int <b>Value:2</b>
Constants.FV_UpdateAllManualClientTi	int <b>Value:3</b>
Constants.FP_LastUpdate	int <b>Value:2052</b>
Constants.FP_TiFile	int <b>Value:2053</b>
Constants.FP_TiMacEdition	int <b>Value:2054</b>
Constants.FP_ImportHint	int <b>Value:2055</b>
Constants.FP_TiLocked	int <b>Value:2056</b>
Constants.FP_TiFileModDate	int <b>Value:2057</b>
Constants.FP_TiIsNested	int <b>Value:2058</b>
Constants.FP_TiMainFlow	int <b>Value:2059</b>
Constants.FP_TiFlowName	int <b>Value:2060</b>
Constants.FP_TiFlowPageSpace	int <b>Value:2061</b>
Constants.FV_BODY_PAGE	int <b>Value:0x00</b>
Constants.FV_REFERENCE_PAGE	int <b>Value:0x02</b>
Constants.FP_TiFormat	int <b>Value:2062</b>
Constants.FP_TiRemovePageBreaks	int <b>Value:2063</b>

Property name	Data Type
Constants.FP_TiRemoveOverrides	int <b>Value:2064</b>
Constants.FP_TiTextEncoding	int <b>Value:2073</b>
Constants.FV_IsoLatin	int <b>Value:1</b>
Constants.FV_ASCII	int <b>Value:2</b>
Constants.FV_ANSI	int <b>Value:3</b>
Constants.FV_MacANSI	int <b>Value:4</b>
Constants.FV_JIS	int <b>Value:5</b>
Constants.FV_Shift_JIS	int <b>Value:6</b>
Constants.FV_EUC	int <b>Value:7</b>
Constants.FV_BIG5	int <b>Value:8</b>
Constants.FV_EUC_CNS	int <b>Value:9</b>
Constants.FV_GB	int <b>Value:10</b>
Constants.FV_HZ	int <b>Value:11</b>
Constants.FV_Korean	int <b>Value:12</b>
Constants.FP_TiEOLisEOP	int <b>Value:2065</b>
Constants.FP_TiTblTag	int <b>Value:2066</b>
Constants.FP_TiNumHeaderRows	int <b>Value:2067</b>

Property name	Data Type
Constants.FP_TiHeadersEmpty	int <b>Value:</b> 2068
Constants.FP_TiByRows	int <b>Value:</b> 2069
Constants.FP_TiSeparator	int <b>Value:</b> 2070
Constants.FP_TiNumSeparators	int <b>Value:</b> 2071
Constants.FP_TiNumCols	int <b>Value:</b> 2072
Constants.FP_TiIsUnresolved	int <b>Value:</b> 2075
Constants.FP_TiClientName	int <b>Value:</b> 2076
Constants.FP_TiClientSource	int <b>Value:</b> 2077
Constants.FP_TiClientType	int <b>Value:</b> 2078
Constants.FP_TiClientData	int <b>Value:</b> 2079
Constants.FP_TiClientUpdateSilent	int <b>Value:</b> 2080
Constants.FP_ContentHeight	int <b>Value:</b> 2219
Constants.FP_ParentTextFrame	int <b>Value:</b> 2200
Constants.FP_PrevSubCol	int <b>Value:</b> 2201
Constants.FP_NextSubCol	int <b>Value:</b> 2202
Constants.FP_Overflowed	int <b>Value:</b> 983
Constants.FP_FirstCombinedFontDefnIn Doc	int <b>Value:</b> 2280

Property name	Data Type
Constants.FP_NextCombinedFontDefnInDoc	int <b>Value:2281</b>
Constants.FP_BaseFamily	int <b>Value:2282</b>
Constants.FP_AllowBaseFamilyBoldedAndObliqued	int <b>Value:2283</b>
Constants.FP_WesternFamily	int <b>Value:2284</b>
Constants.FP_WesternSize	int <b>Value:2285</b>
Constants.FP_WesternShift	int <b>Value:2286</b>
Constants.FP_CombinedFontFamily	int <b>Value:2287</b>
Constants.FS_FindText	int <b>Value:1</b>
Constants.FS_FindElementTag	int <b>Value:2</b>
Constants.FV_FindElemTag	int <b>Value:0</b>
Constants.FV_FindAttrName	int <b>Value:1</b>
Constants.FV_FindAttrValue	int <b>Value:2</b>
Constants.FV_NumFindElementItems	int <b>Value:2+1</b>
Constants.FS_FindCharFmt	int <b>Value:3</b>
Constants.FS_FindPgftag	int <b>Value:4</b>
Constants.FS_FindCharTag	int <b>Value:5</b>
Constants.FS_FindTableTag	int <b>Value:6</b>

Property name	Data Type
Constants.FS_FindObject	int <b>Value:8</b>
Constants.FV_FindAnyMarker	int <b>Value:0</b>
Constants.FV_FindAnyXRef	int <b>Value:1</b>
Constants.FV_FindUnresolvedXRef	int <b>Value:2</b>
Constants.FV_FindAnyTextInset	int <b>Value:3</b>
Constants.FV_FindUnresolvedTextInset	int <b>Value:4</b>
Constants.FV_FindAnyPub	int <b>Value:5</b>
Constants.FV_FindAnyVariable	int <b>Value:6</b>
Constants.FV_FindAnchoredFrame	int <b>Value:7</b>
Constants.FV_FindFootnote	int <b>Value:8</b>
Constants.FV_FindAnyTable	int <b>Value:9</b>
Constants.FV_FindAutomaticHyphen	int <b>Value:10</b>
Constants.FV_FindAnyRubi	int <b>Value:11</b>
Constants.FV_FindPgFormatOverride	int <b>Value:12</b>
Constants.FV_FindCharacterFormatOverride	int <b>Value:13</b>
Constants.FV_FindTableFormatOverride	int <b>Value:14</b>
Constants.FS_FindMarkerOfType	int <b>Value:9</b>

Property name	Data Type
Constants.FS_FindMarkerText	int <b>Value:10</b>
Constants.FS_FindXRefWithFormat	int <b>Value:11</b>
Constants.FS_FindNamedVariable	int <b>Value:12</b>
Constants.FS_FindCondTextInCondTags	int <b>Value:13</b>
Constants.FS_FindCondTextNotInCondTags	int <b>Value:14</b>
Constants.FS_FindCustomizationFlags	int <b>Value:15</b>
Constants.FS_FindWrap	int <b>Value:16</b>
Constants.FS_SpellOptions	int <b>Value:1000</b>
Constants.FS_TwoInARowString	int <b>Value:1001</b>
Constants.FS_TextBeforeString	int <b>Value:1002</b>
Constants.FS_TextAfterString	int <b>Value:1003</b>
Constants.FS_WordContainString	int <b>Value:1004</b>
Constants.FS_SpellAction	int <b>Value:1005</b>
Constants.FV_CheckDocument	int <b>Value:1</b>
Constants.FV_CheckCurrentPage	int <b>Value:2</b>
Constants.FV_WriteUnknownWordsToFile	int <b>Value:3</b>
Constants.FS_OutputFilePathName	int <b>Value:1006</b>

Property name	Data Type
Constants.FS_SpellWrap	int <b>Value:</b> 1007
Constants.FS_NumSpellParams	int <b>Value:</b> (1007-1000+1)
Constants.FV_SpellMisspelling	int <b>Value:</b> 1
Constants.FV_SpellCapitalization	int <b>Value:</b> 2
Constants.FV_SpellRepeatedWord	int <b>Value:</b> 3
Constants.FV_SpellRepeatedLetter	int <b>Value:</b> 4
Constants.FV_SpellHyphenation	int <b>Value:</b> 5
Constants.FV_SpellExtraSpace	int <b>Value:</b> 6
Constants.FV_SpellSpaceAfter	int <b>Value:</b> 7
Constants.FV_SpellSpaceBefore	int <b>Value:</b> 8
Constants.FV_SpellStraightQuotes	int <b>Value:</b> 9
Constants.FV_UserDictionary	int <b>Value:</b> 1
Constants.FV_DocumentDictionary	int <b>Value:</b> 2
Constants.FV_WriteDictionaryToFile	int <b>Value:</b> 4
Constants.FV_MergeDictionaryContents	int <b>Value:</b> 5
Constants.FV_SetDictionaryToNoneDictionary	int <b>Value:</b> 6
Constants.FV_ClearDocDictionary	int <b>Value:</b> 7



Property name	Data Type
Constants.FV_ChangePersonalDictionary	int <b>Value:8</b>
Constants.FV_GetDictionaryStrings	int <b>Value:9</b>
Constants.FP_CursorData	int <b>Value:2294</b>
Constants.FP_CursorTypes	int <b>Value:2295</b>
Constants.FP_StructureCursorTypes	int <b>Value:2296</b>
Constants.FP_SecNumComputeMethod	int <b>Value:915</b>
Constants.FP_SectionNumStyle	int <b>Value:916</b>
Constants.FP_SectionNumber	int <b>Value:917</b>
Constants.FP_SectionNumText	int <b>Value:918</b>
Constants.FP_SubsecNumComputeMethod	int <b>Value:815</b>
Constants.FP_SubsectionNumStyle	int <b>Value:816</b>
Constants.FP_SubsectionNumber	int <b>Value:817</b>
Constants.FP_SubsectionNumText	int <b>Value:818</b>
Constants.FP_DoNotGenerateErrorLog	int <b>Value:819</b>
Constants.FP_DocOpenClientEncounteredErrors	int <b>Value:820</b>
Constants.FP_OpenAndSaveXmlBookComponentDoc	int <b>Value:822</b>
Constants.FP_XmlIsBook	int <b>Value:823</b>

Property name	Data Type
Constants.FP_UseFMConsoleForErrorLog	int <b>Value:</b> 825
Constants.FP_FMConsoleString	int <b>Value:</b> 826
Constants.FP_IsTempOpenSave	int <b>Value:</b> 827
Constants.FP_AllowNewFileURL	int <b>Value:</b> 2416
Constants.FP_DoPostXSLTValidationOnExport	int <b>Value:</b> 2417
Constants.FP_DoNotExportInvalidXML	int <b>Value:</b> 2418
Constants.FP_SuppressXMLParserWarnings	int <b>Value:</b> 2419
Constants.FP_RemoveExtraWhiteSpacesOnXMLImport	int <b>Value:</b> 2830
Constants.FP_NoMultiMediaInPDF	int <b>Value:</b> 2831
Constants.FP_InsetGenericData	int <b>Value:</b> 2832
Constants.FP_InTableSortOperation	int <b>Value:</b> 2833
Constants.FP_InsetGfxName	int <b>Value:</b> 2834
Constants.FP_InsetGfxPlayWindowInPdf	int <b>Value:</b> 2835
Constants.FP_InsetGfxActiveInPdf	int <b>Value:</b> 2836
Constants.FP_InsetJavaScriptAttached	int <b>Value:</b> 2837
Constants.FP_InsetJavaScriptFile	int <b>Value:</b> 2838
Constants.FP_InsetMonikerPath	int <b>Value:</b> 2848

Property name	Data Type
Constants.FP_InsetMonikerFilePath	int <b>Value:2849</b>
Constants.FP_IsFateRun	int <b>Value:2847</b>
Constants.FP_DefaultVectorFormatForXMLExport	int <b>Value:2874</b>
Constants.FV_CmdOpenXmlElementId	int <b>Value:821</b>
Constants.FV_FileTypeBinaryDoc	int <b>Value:1</b>
Constants.FV_FileTypeBinaryBook	int <b>Value:2</b>
Constants.FV_FileTypeMif	int <b>Value:3</b>
Constants.FV_FileTypeXml	int <b>Value:4</b>
Constants.FV_FileTypeSgml	int <b>Value:5</b>
Constants.FV_FileTypeText	int <b>Value:6</b>
Constants.FV_FileTypeFilter	int <b>Value:7</b>
Constants.FV_FileTypeFolder	int <b>Value:8</b>
Constants.FV_FileTypeGeneral	int <b>Value:9</b>
Constants.FV_FileTypeErrorDoc	int <b>Value:10</b>
Constants.FV_FileTypeEquation	int <b>Value:11</b>
Constants.FV_FileTypeThesaurus	int <b>Value:12</b>
Constants.FV_FileTypeCompositeDoc	int <b>Value:13</b>

Property name	Data Type
Constants.FV_FileTypeSummaryDoc	int <b>Value:14</b>
Constants.FS_StructuredApplication	int <b>Value:1</b>
Constants.FS_Doctype	int <b>Value:2</b>
Constants.FS_PublicId	int <b>Value:3</b>
Constants.FS_SystemId	int <b>Value:4</b>
Constants.FS_Extension	int <b>Value:5</b>
Constants.FS_Visible	int <b>Value:6</b>
Constants.FS_NumNewXMLParams	int <b>Value:6</b>
Constants.FV_ListViewInit	int <b>Value:1</b>
Constants.FV_ListViewAddColumn	int <b>Value:2</b>
Constants.FV_ListViewAddRow	int <b>Value:3</b>
Constants.FV_ListViewDeleteAllRows	int <b>Value:4</b>
Constants.FV_ListViewReSortAllRows	int <b>Value:5</b>
Constants.FV_ListViewSetSortEnabled	int <b>Value:6</b>
Constants.FV_ListViewSelectFirstRow	int <b>Value:7</b>
Constants.FV_ListViewSelectRowFromCo lData	int <b>Value:8</b>
Constants.FV_ListViewSelectRowFromRo wID	int <b>Value:9</b>

Property name	Data Type
Constants.FV_ListViewActionMax	int <b>Value:10</b>
Constants.FV_ListViewInitNumParams	int <b>Value:0</b>
Constants.FV_ListViewAddColumnNumParams	int <b>Value:4</b>
Constants.FV_ListViewAddColumnParam1	int <b>Value:3</b>
Constants.FV_ListViewAddColumnParam2	int <b>Value:1</b>
Constants.FV_ListViewAddColumnParam3	int <b>Value:1</b>
Constants.FV_ListViewAddColumnParam4	int <b>Value:1</b>
Constants.FV_ListViewAddRowNumParams	int <b>Value:1</b>
Constants.FV_ListViewAddRowParam1	int <b>Value:6</b>
Constants.FV_ListViewDeleteAllRowsNumParams	int <b>Value:0</b>
Constants.FV_ListViewReSortAllRowsNumParams	int <b>Value:0</b>
Constants.FP_LineNumDistance	int <b>Value:4001</b>
Constants.FP_LineNumRestart	int <b>Value:4002</b>
Constants.FP_LineNumShow	int <b>Value:4003</b>
Constants.FP_LineNumFont	int <b>Value:4004</b>
Constants.FP_LineNumSize	int <b>Value:4005</b>
Constants.FP_LineNumColor	int <b>Value:4006</b>

Property name	Data Type
Constants.FP_IsDefault	int <b>Value:2850</b>
Constants.FP_IsStale	int <b>Value:2851</b>
Constants.FP_NextKeyCatalogInSession	int <b>Value:2852</b>
Constants.FP_NotLoaded	int <b>Value:2853</b>
Constants.FP_Source	int <b>Value:2854</b>
Constants.FP_SourceType	int <b>Value:2855</b>
Constants.FP_KeyCount	int <b>Value:2860</b>
Constants.FP_KeyCatalogClientName	int <b>Value:2873</b>
Constants.FV_KeySrcTypeNone	int <b>Value:0</b>
Constants.FV_KeySrcTypeDitamap	int <b>Value:1</b>
Constants.FV_DocKeyCatalogTypeDefault	int <b>Value:0</b>
Constants.FV_DocKeyCatalogTypeSpecified	int <b>Value:1</b>
Constants.FV_DocKeyCatalogTypeNone	int <b>Value:2</b>
Constants.FV_DocKeyCatalogTypeMax	int <b>Value:3</b>
Constants.FV_KeydefVariableInfoLen	int <b>Value:3</b>
Constants.FV_KeydefVariableInfoTagPrimary	int <b>Value:0</b>
Constants.FV_KeydefVariableInfoTagSecondary	int <b>Value:1</b>

Property name	Data Type
Constants.FV_KeydefVariableInfoText	int <b>Value:2</b>
Constants.FV_KeyDefFieldsTypePrimary	int <b>Value:0</b>
Constants.FV_KeyDefFieldsTypeAll	int <b>Value:1</b>
Constants.FV_KeyDefFieldsTypeMax	int <b>Value:2</b>
Constants.FV_KeydefKeyTag	int <b>Value:1</b>
Constants.FV_KeydefKeyTarget	int <b>Value:2</b>
Constants.FV_KeydefKeySrcFile	int <b>Value:3</b>
Constants.FV_KeydefKeyDuplicate	int <b>Value:4</b>
Constants.FV_KeyDefMaxPrimaryFields	int <b>Value:4+1</b>
Constants.FV_KeydefKeySrcType	int <b>Value:5</b>
Constants.FV_KeydefKeyVarList	int <b>Value:6</b>
Constants.FV_KeydefKeyDefaultText	int <b>Value:7</b>
Constants.FV_KeydefKeyFoundInRefFile	int <b>Value:8</b>
Constants.FV_KeydefKeyInvalid	int <b>Value:9</b>
Constants.FV_KeydefKeyAttrs	int <b>Value:10</b>
Constants.FV_KeyDefMaxFields	int <b>Value:11</b>
Constants.FV_KeydefKeyTagType	int <b>Value:3</b>

Property name	Data Type
Constants.FV_KeydefKeyTargetType	int <b>Value:3</b>
Constants.FV_KeydefKeySrcFileType	int <b>Value:3</b>
Constants.FV_KeydefKeyDuplicateType	int <b>Value:1</b>
Constants.FV_KeydefKeySrcTypeType	int <b>Value:1</b>
Constants.FV_KeydefKeyVarListType	int <b>Value:20</b>
Constants.FV_KeydefKeyDefaultTextType	int <b>Value:3</b>
Constants.FV_KeydefKeyFoundInRefFileType	int <b>Value:1</b>
Constants.FV_KeydefKeyInvalidType	int <b>Value:1</b>
Constants.FV_KeydefKeyAttrsType	int <b>Value:21</b>
Constants.FV_KeyDefFieldSize	int <b>Value:2</b>
Constants.FF_DUPLICATE_KEY_DEFINITION	int <b>Value:0x01</b>
Constants.FF_FOUND_IN_REFERENCED_FILE	int <b>Value:0x02</b>
Constants.FF_INVALID_KEY	int <b>Value:0x04</b>
Constants.FV_DITAObjTypeAuto	int <b>Value:0</b>
Constants.FV_DITAObjTypeConref	int <b>Value:1</b>
Constants.FV_DITAObjTypeXref	int <b>Value:2</b>
Constants.FV_DITAObjTypeLink	int <b>Value:3</b>



Property name	Data Type
Constants.FV_DITAObjTypeTopicref	int <b>Value:4</b>
Constants.FV_DITAObjTypeTopicsetref	int <b>Value:5</b>
Constants.FV_DITAObjTypeMax	int <b>Value:6</b>
Constants.FF_DITAUpdateAllConrefs	int <b>Value:0x01</b>
Constants.FF_DITAUpdateAllXrefs	int <b>Value:0x02</b>
Constants.FF_DITAUpdateAllLinks	int <b>Value:0x04</b>
Constants.FF_DITAUpdateAllTopicrefs	int <b>Value:0x08</b>
Constants.FF_DITAUpdateAllTopicsetrefs	int <b>Value:0x10</b>
Constants.FF_DITAUpdateAllReferences	int <b>Value:(0x01   0x02   0x04   0x08   0x10)</b>
Constants.FP_FirstDITAConrefElementInDoc	int <b>Value:2861</b>
Constants.FP_FirstDITAXrefElementInDoc	int <b>Value:2862</b>
Constants.FP_FirstDITALinkElementInDoc	int <b>Value:2863</b>
Constants.FP_FirstDITATopicrefElementInDoc	int <b>Value:2864</b>
Constants.FP_FirstDITATopicsetrefElementInDoc	int <b>Value:2865</b>
Constants.FP_NextDITAConrefElementInDoc	int <b>Value:2866</b>
Constants.FP_NextDITAXrefElementInDoc	int <b>Value:2867</b>
Constants.FP_NextDITALinkElementInDoc	int <b>Value:2868</b>

Property name	Data Type
Constants.FP_NextDITATopicrefElementInDoc	int <b>Value:</b> 2869
Constants.FP_NextDITATopicsetrefElementInDoc	int <b>Value:</b> 2870
Constants.FV_DITA_TYPE_GENERAL	int <b>Value:</b> 1
Constants.FV_DITA_TYPE_RELTABLE	int <b>Value:</b> 2
Constants.FV_DITA_TYPE_KEYDEF	int <b>Value:</b> 3
Constants.FV_DITA_TYPE_TOPICGROUP	int <b>Value:</b> 4
Constants.FV_DITA_TYPE_TOPICHEAD	int <b>Value:</b> 5
Constants.FV_DITA_TYPE_TOPICSETREF	int <b>Value:</b> 6
Constants.FV_DITA_TYPE_TOPICSET	int <b>Value:</b> 7
Constants.FV_DITA_TYPE_MAPREF	int <b>Value:</b> 8
Constants.FV_DITA_TYPE_NAVREF	int <b>Value:</b> 9
Constants.FV_DITA_TYPE_TITLE	int <b>Value:</b> 10
Constants.FV_DITA_TYPE_TOPICREF	int <b>Value:</b> 11
Constants.FV_DITA_TYPE_ANCHOR	int <b>Value:</b> 12
Constants.FV_ENUMERATE_DITACOMP_TOPICREF	int <b>Value:</b> 1 << 1
Constants.FV_ENUMERATE_DITACOMP_MAPREF	int <b>Value:</b> 1 << 2
Constants.FV_ENUMERATE_DITACOMP_TOPICSET	int <b>Value:</b> 1 << 3

Property name	Data Type
Constants.FV_ENUMERATE_DITACOMP_TOPICSETREF	int <b>Value:</b> 1 << 4
Constants.FV_ENUMERATE_DITACOMP_KEYDEF	int <b>Value:</b> 1 << 5
Constants.FV_ENUMERATE_COMP_CHILDMAP_COMPONENTS	int <b>Value:</b> 1 << 6
Constants.FV_ENUMERATE_COMP_CHILDBOOK_COMPONENTS	int <b>Value:</b> 1 << 7
Constants.FV_ENUMERATE_BOOKCOMP_FOLDER	int <b>Value:</b> 1 << 8
Constants.FV_ENUMERATE_BOOKCOMP_GROUP	int <b>Value:</b> 1 << 9
Constants.FV_ENUMERATE_COMP_FOLDER_COMPONENTS	int <b>Value:</b> 1 << 10
Constants.FV_ENUMERATE_COMP_GROUP_COMPONENTS	int <b>Value:</b> 1 << 11
Constants.FV_ENUMERATE_COMPFILE_FMBINARY	int <b>Value:</b> 1 << 12
Constants.FV_ENUMERATE_COMPFILE_FMMIF	int <b>Value:</b> 1 << 13
Constants.FV_ENUMERATE_COMPFILE_BOOKBINARY	int <b>Value:</b> 1 << 14
Constants.FV_ENUMERATE_COMPFILE_BOOKMIF	int <b>Value:</b> 1 << 15
Constants.FV_ENUMERATE_COMPFILE_XML	int <b>Value:</b> 1 << 16
Constants.FMV_CMDSUPPORTED	int <b>Value:</b> -1
Constants.FF_UNTAGGED_ASK	int <b>Value:</b> 0
Constants.FF_UNTAGGED_UNCOND	int <b>Value:</b> 1
Constants.FF_UNTAGGED_DELETE	int <b>Value:</b> 2

Property name	Data Type
Constants.FS_IteratorItemFieldBookId	int <b>Value:1</b>
Constants.FS_IteratorItemFieldDocId	int <b>Value:2</b>
Constants.FS_IteratorItemFieldObjId	int <b>Value:3</b>
Constants.FS_IteratorItemMaxFields	int <b>Value:3</b>
Constants.FS_IterParamParseBookInBook	int <b>Value:1</b>
Constants.FS_IterParamParseMapInBook	int <b>Value:2</b>
Constants.FS_IterParamSkipExcludedBookComponents	int <b>Value:3</b>
Constants.FS_IterParamGetBookComponentFolder	int <b>Value:4</b>
Constants.FS_IterParamGetBookComponentGroup	int <b>Value:5</b>
Constants.FS_IterParamGetBookComponentRefBook	int <b>Value:6</b>
Constants.FS_IterParamGetBookComponentRefMap	int <b>Value:7</b>
Constants.FS_IterParamParseMapInMap	int <b>Value:8</b>
Constants.FS_IterParamSkipResourceOnlyTopicrefs	int <b>Value:9</b>
Constants.FS_IterParamSkipNonPrintTopicrefs	int <b>Value:10</b>
Constants.FS_IterParamSkipTopicsetrefs	int <b>Value:11</b>
Constants.FS_IterParamGetMapRef	int <b>Value:12</b>
Constants.FS_NumIteratorParams	int <b>Value:12</b>

Property name	Data Type
Constants.FV_CMSOpNone	int <b>Value:</b> FV_CMSOpNone
Constants.FV_CMSOpItemUpdated	int <b>Value:</b> FV_CMSOpItemUpdated
Constants.FV_CMSOpDependentsUpdated	int <b>Value:</b> FV_CMSOpDependentsUpdated
Constants.FV_CMSOpDependentsDeleted	int <b>Value:</b> FV_CMSOpDependentsDeleted
Constants.FV_CMSOpItemDeleted	int <b>Value:</b> FV_CMSOpItemDeleted
Constants.FV_CMSOpChildAdded	int <b>Value:</b> FV_CMSOpChildAdded
Constants.FV_CMSOpRootUpdated	int <b>Value:</b> FV_CMSOpRootUpdated
Constants.FP_CMSItemProperty_Min	int <b>Value:</b> FP_CMSItemProperty_Min
Constants.FP_CMSItemProperty_ItemName	int <b>Value:</b> FP_CMSItemProperty_ItemName
Constants.FP_CMSItemProperty_ItemServerPath	int <b>Value:</b> FP_CMSItemProperty_ItemServerPath
Constants.FP_CMSItemProperty_ItemLocalPath	int <b>Value:</b> FP_CMSItemProperty_ItemLocalPath
Constants.FP_CMSItemProperty_ItemsIsCheckedOut	int <b>Value:</b> FP_CMSItemProperty_ItemsIsCheckedOut
Constants.FP_CMSItemProperty_ItemCheckedOutByCurrentUser	int <b>Value:</b> FP_CMSItemProperty_ItemCheckedOutByCurrentUser
Constants.FP_CMSItemProperty_ItemsIsContainer	int <b>Value:</b> FP_CMSItemProperty_ItemsIsContainer
Constants.FP_CMSItemProperty_ItemType	int <b>Value:</b> FP_CMSItemProperty_ItemType
Constants.FP_CMSItemProperty_ItemFileType	int <b>Value:</b> FP_CMSItemProperty_ItemFileType

Property name	Data Type
Constants.FP_CMSItemProperty_ItemVersion	int <b>Value:</b> FP_CMSItemProperty_ItemVersion
Constants.FP_CMSItemProperty_Max	int <b>Value:</b> FP_CMSItemProperty_Max
Constants.FV_CMSItemPropertyValue_Min	int <b>Value:</b> FV_CMSItemPropertyValue_Min
Constants.FV_CMSItemPropertyValue_Root	int <b>Value:</b> FV_CMSItemPropertyValue_Root
Constants.FV_CMSItemPropertyValue_Folder	int <b>Value:</b> FV_CMSItemPropertyValue_Folder
Constants.FV_CMSItemPropertyValue_File	int <b>Value:</b> FV_CMSItemPropertyValue_File
Constants.FV_CMSItemPropertyValue_General	int <b>Value:</b> FV_CMSItemPropertyValue_General
Constants.FV_CMSItemPropertyValue_Max	int <b>Value:</b> FV_CMSItemPropertyValue_Max
Constants.FV_CMSItemFileTypeValue_Min	int <b>Value:</b> FV_CMSItemFileTypeValue_Min
Constants.FV_CMSItemFileTypeValue_Xml	int <b>Value:</b> FV_CMSItemFileTypeValue_Xml
Constants.FV_CMSItemFileTypeValue_FmDoc	int <b>Value:</b> FV_CMSItemFileTypeValue_FmDoc
Constants.FV_CMSItemFileTypeValue_Mif	int <b>Value:</b> FV_CMSItemFileTypeValue_Mif
Constants.FV_CMSItemFileTypeValue_DitaMap	int <b>Value:</b> FV_CMSItemFileTypeValue_DitaMap
Constants.FV_CMSItemFileTypeValue_DitaTopic	int <b>Value:</b> FV_CMSItemFileTypeValue_DitaTopic
Constants.FV_CMSItemFileTypeValue_FmBook	int <b>Value:</b> FV_CMSItemFileTypeValue_FmBook
Constants.FV_CMSItemFileTypeValue_Text	int <b>Value:</b> FV_CMSItemFileTypeValue_Text
Constants.FV_CMSItemFileTypeValue_Img	int <b>Value:</b> FV_CMSItemFileTypeValue_Img

Property name	Data Type
Constants.FV_CMSItemFileTypeValue_General	int <b>Value:</b> FV_CMSItemFileTypeValue_General
Constants.FV_CMSItemFileTypeValue_Max	int <b>Value:</b> FV_CMSItemFileTypeValue_Max
Constants.FA_CMSCCommandMin	int <b>Value:</b> FA_CMSCCommandMin
Constants.FA_CMSCreateConnection	int <b>Value:</b> FA_CMSCreateConnection
Constants.FA_CMSSetRootObject	int <b>Value:</b> FA_CMSSetRootObject
Constants.FA_CMSCreateConnMgrUI	int <b>Value:</b> FA_CMSCreateConnMgrUI
Constants.FA_CMSGetItemFromPath	int <b>Value:</b> FA_CMSGetItemFromPath
Constants.FA_CMSCloseConnection	int <b>Value:</b> FA_CMSCloseConnection
Constants.FA_CMSGetCommandMaxOpCode	int <b>Value:</b> FA_CMSGetCommandMaxOpCode
Constants.FA_CMSGetPropertyMaxOpCode	int <b>Value:</b> FA_CMSGetPropertyMaxOpCode
Constants.FA_CMSObjectCheckout	int <b>Value:</b> FA_CMSObjectCheckout
Constants.FA_CMSObjectCheckin	int <b>Value:</b> FA_CMSObjectCheckin
Constants.FA_CMSObjectCancelCheckout	int <b>Value:</b> FA_CMSObjectCancelCheckout
Constants.FA_CMSObjectEdit	int <b>Value:</b> FA_CMSObjectEdit
Constants.FA_CMSObjectOpenReadOnly	int <b>Value:</b> FA_CMSObjectOpenReadOnly
Constants.FA_CMSObjectDelete	int <b>Value:</b> FA_CMSObjectDelete
Constants.FA_CMSObjectShowVersion	int <b>Value:</b> FA_CMSObjectShowVersion

Property name	Data Type
Constants.FA_CMSObjectShowDependents	int <b>Value:</b> FA_CMSObjectShowDependents
Constants.FA_CMSObjectShowProperties	int <b>Value:</b> FA_CMSObjectShowProperties
Constants.FA_CMSObjectShowCheckOutFiles	int <b>Value:</b> FA_CMSObjectShowCheckOutFiles
Constants.FA_CMSObjectDownload	int <b>Value:</b> FA_CMSObjectDownload
Constants.FA_CMSObjectDownloadItem	int <b>Value:</b> FA_CMSObjectDownloadItem
Constants.FA_CMSObjectDownloadForOpen	int <b>Value:</b> FA_CMSObjectDownloadForOpen
Constants.FA_CMSObjectUploadFile	int <b>Value:</b> FA_CMSObjectUploadFile
Constants.FA_CMSObjectUploadFolder	int <b>Value:</b> FA_CMSObjectUploadFolder
Constants.FA_CMSObjectGetChildren	int <b>Value:</b> FA_CMSObjectGetChildren
Constants.FA_CMSObjectIsSame	int <b>Value:</b> FA_CMSObjectIsSame
Constants.FA_CMSObjectRefresh	int <b>Value:</b> FA_CMSObjectRefresh
Constants.FA_CMSSimpleSearch	int <b>Value:</b> FA_CMSSimpleSearch
Constants.FA_CMSAdvancedSearch	int <b>Value:</b> FA_CMSAdvancedSearch
Constants.FA_CMSGetItems	int <b>Value:</b> FA_CMSGetItems
Constants.FA_CMSBuildContextMenu	int <b>Value:</b> FA_CMSBuildContextMenu
Constants.FA_CMSIsValidCommand	int <b>Value:</b> FA_CMSIsValidCommand
Constants.FA_CMSCCommandMax	int <b>Value:</b> FA_CMSCCommandMax



Property name	Data Type
Constants.FV_CMSTMenu_Min	int <b>Value:</b> FV_CMSTMenu_Min
Constants.FV_CMSTMenu_Is_Item	int <b>Value:</b> FV_CMSTMenu_Is_Item
Constants.FV_CMSTMenu_Is_Disabled	int <b>Value:</b> FV_CMSTMenu_Is_Disabled
Constants.FV_CMSTMenu_Is_Separator	int <b>Value:</b> FV_CMSTMenu_Is_Separator
Constants.FV_CMSTMenu_Is_SubMenu	int <b>Value:</b> FV_CMSTMenu_Is_SubMenu
Constants.FV_CMSTMenu_Max	int <b>Value:</b> FV_CMSTMenu_Max
Constants.FV_CMSVersion_Min	int <b>Value:</b> FV_CMSVersion_Min
Constants.FV_CMSSameVersion	int <b>Value:</b> FV_CMSSameVersion
Constants.FV_CMSMinorVersion	int <b>Value:</b> FV_CMSMinorVersion
Constants.FV_CMSMajorVersion	int <b>Value:</b> FV_CMSMajorVersion
Constants.FV_CMSVersion_Max	int <b>Value:</b> FV_CMSVersion_Max
Constants.FV_CMSCheckoutUI_Id_Show Dependents	int <b>Value:</b> FV_CMSCheckoutUI_Id_ShowDependents
Constants.FV_CMSCheckinUI_Id_SameVersion	int <b>Value:</b> FV_CMSCheckinUI_Id_SameVersion
Constants.FV_CMSCheckinUI_Id_MinorVersion	int <b>Value:</b> FV_CMSCheckinUI_Id_MinorVersion
Constants.FV_CMSCheckinUI_Id_MajorVersion	int <b>Value:</b> FV_CMSCheckinUI_Id_MajorVersion
Constants.FV_CMSCheckinUI_Id_Version Label	int <b>Value:</b> FV_CMSCheckinUI_Id_VersionLabel
Constants.FV_CMSCheckinUI_Id_Description	int <b>Value:</b> FV_CMSCheckinUI_Id_Description

Property name	Data Type
Constants.FV_CMSCheckinUI_Id_CheckinComment	int <b>Value:</b> FV_CMSCheckinUI_Id_CheckinComment
Constants.FV_CMSCheckinUI_Id_MakeThisCurrentVersion	int <b>Value:</b> FV_CMSCheckinUI_Id_MakeThisCurrentVersion
Constants.FV_CMDeleteUI_DeleteAllVersion	int <b>Value:</b> FV_CMDeleteUI_DeleteAllVersion
Constants.FV_CMDeleteUI_DeleteAllDependents	int <b>Value:</b> FV_CMDeleteUI_DeleteAllDependents
Constants.FV_CMSCommandId_Min	int <b>Value:</b> FV_CMSCommandId_Min
Constants.FV_CMSCommandNameId	int <b>Value:</b> FV_CMSCommandNameId
Constants.FV_CMSCommandConnTypeId	int <b>Value:</b> FV_CMSCommandConnTypeId
Constants.FV_CMSCommandServerId	int <b>Value:</b> FV_CMSCommandServerId
Constants.FV_CMSCommandUserNameId	int <b>Value:</b> FV_CMSCommandUserNameId
Constants.FV_CMSCommandPasswordId	int <b>Value:</b> FV_CMSCommandPasswordId
Constants.FV_CMSCommandUserField1	int <b>Value:</b> FV_CMSCommandUserField1
Constants.FV_CMSCommandRepositoryId	int <b>Value:</b> FV_CMSCommandRepositoryId
Constants.FV_CMSCommandUserField2	int <b>Value:</b> FV_CMSCommandUserField2
Constants.FV_CMSCommandFilePathId	int <b>Value:</b> FV_CMSCommandFilePathId
Constants.FV_CMSCommandSearchStringId	int <b>Value:</b> FV_CMSCommandSearchStringId
Constants.FV_CMSCommandAdvancedSearchStringId	int <b>Value:</b> FV_CMSCommandAdvancedSearchStringId
Constants.FV_CMSCommandId	int <b>Value:</b> FV_CMSCommandId

Property name	Data Type
Constants.FV_CMSSContextMenuId	int <b>Value:</b> FV_CMSSContextMenuId
Constants.FV_CMSSContextMenuString	int <b>Value:</b> FV_CMSSContextMenuString
Constants.FV_CMSSilentOperation	int <b>Value:</b> FV_CMSSilentOperation
Constants.FV_CMSSCommandCheckoutWithDescendentId	int <b>Value:</b> FV_CMSSCommandCheckoutWithDescendentId
Constants.FV_CMSSCommandCheckinMakeCurrentVersionId	int <b>Value:</b> FV_CMSSCommandCheckinMakeCurrentVersionId
Constants.FV_CMSSCommandCheckinKeepLocalCopyId	int <b>Value:</b> FV_CMSSCommandCheckinKeepLocalCopyId
Constants.FV_CMSSCommandCheckinMinorVersionId	int <b>Value:</b> FV_CMSSCommandCheckinMinorVersionId
Constants.FV_CMSSCommandCheckinVersionLabelId	int <b>Value:</b> FV_CMSSCommandCheckinVersionLabelId
Constants.FV_CMSSCommandCheckinDescriptionId	int <b>Value:</b> FV_CMSSCommandCheckinDescriptionId
Constants.FV_CMSSCommandCheckinCommentId	int <b>Value:</b> FV_CMSSCommandCheckinCommentId
Constants.FV_CMSSCommandDeleteAllVersionId	int <b>Value:</b> FV_CMSSCommandDeleteAllVersionId
Constants.FV_CMSSCommandDeleteAllDependentsId	int <b>Value:</b> FV_CMSSCommandDeleteAllDependentsId
Constants.FV_CMSSCommandOpenReadOnlyId	int <b>Value:</b> FV_CMSSCommandOpenReadOnlyId
Constants.FV_CMSSCommandSilentOpenId	int <b>Value:</b> FV_CMSSCommandSilentOpenId
Constants.FV_CMSSCommandId_Max	int <b>Value:</b> FV_CMSSCommandId_Max
Constants.FE_CMSSNameAlreadyRegistered	int <b>Value:</b> -112
Constants.FE_CMSSBadSessionId	int <b>Value:</b> -113

Property name	Data Type
Constants.FE_CMSBadObjectId	int <b>Value:-114</b>
Constants.FE_CMSSessionFailed	int <b>Value:-115</b>
Constants.FE_CMSBadCommandId	int <b>Value:-116</b>
Constants.FE_CMSObjectCreationFailed	int <b>Value:-117</b>
Constants.FE_CMSRootObjectExists	int <b>Value:-118</b>
Constants.FE_CMSBadItemFileType	int <b>Value:-119</b>
Constants.FE_CMSBadItemType	int <b>Value:-120</b>
Constants.FE_CMSBadItemContainerType	int <b>Value:-121</b>
Constants.FE_CMSSessionCreationFailed	int <b>Value:-122</b>
Constants.FE_CMSIsValidCMSCommand	int <b>Value:-123</b>
Constants.FE_CMSFailedLogin	int <b>Value:-150</b>
Constants.FE_CMSFailedCheckout	int <b>Value:-151</b>
Constants.FE_CMSFailedCheckin	int <b>Value:-152</b>
Constants.FE_CMSFailedCancelCheckout	int <b>Value:-153</b>
Constants.FE_CMSFailedDelete	int <b>Value:-154</b>
Constants.FE_CMSFailedOpenFile	int <b>Value:-155</b>
Constants.FE_CMSFailedUploadObject	int <b>Value:-156</b>

Property name	Data Type
Constants.FE_CMSFailedDownloadObject	int Value:-157
Constants.FE_CMSFailedGetItemFrompath	int Value:-158
Constants.FE_CMSFailedLogout	int Value:-159

## Doc

Property name	Data Type	Description
BannerTextDisplay	Bool	Specifies whether banner text should be displayed in a document. Here is an example of usage: <pre>F_ApiGetInt (FV_SessionId, F_ObjHandleT docId, FP_BannerTextDisplay); F_ApiSetInt (FV_SessionId, F_ObjHandleT docId, FP_BannerTextDisplay, True /False);</pre>
CurrentPage	FMOobject	Denotes the current page [Body page, Master page or Reference page] in the document.
FirstBodyPageInDoc	<i>BodyPage</i>	Denotes the first body page in the document.
FirstCharFmtInDoc	<i>CharFmt</i>	Indicates the first character tag in the list of the document's character tags.
FirstColorInDoc	<i>Color</i>	Specifies the first color in the list of document's colors.
FirstCombinedFontDefnInDoc	<i>CombinedFontDefn</i>	Specifies the first combined font definition in the list of the document's combined font definitions.
FirstCondFmtInDoc	<i>CondFmt</i>	Indicates the first condition tag in the list of the document's condition tags.
FirstDITAConrefElementInDoc	string	Gets the first Conref element in the document

Property name	Data Type	Description
FirstDITALinkElementInDoc	<i>string</i>	Gets the first Conref element in the document
FirstDITATopicrefElementInDoc	<i>string</i>	Gets the first Topicref element in the document
FirstDITATopicsetrefElementInDoc	<i>string</i>	Gets the first Topicref element in the document
FirstDITAXrefElementInDoc	<i>string</i>	Gets the first cross reference element in the document
FirstFlowInDoc	<i>Flow</i>	Denotes the first flow in the list of the document's flows.
FirstFnInDoc	<i>Fn</i>	Specifies the first footnote in the list of the document's footnotes.
FirstGraphicInDoc	<i>FMObject</i>	Indicates the first graphic object in the list of the document's graphic objects.
FirstMarkerInDoc	<i>Marker</i>	Denotes the first marker in the list of the document's markers.
FirstMarkerTypeInDoc	<i>MarkerType</i>	Specifies the first marker type in the list of the document's marker types.
FirstMasterPageInDoc	<i>MasterPage</i>	Denotes the first master page in the document.
FirstPgfmtInDoc	<i>Pgfmt</i>	Indicates the first paragraph tag in the list of the document's paragraph tags.
FirstPgInDoc	<i>Pgf</i>	Specifies the first paragraph in the list of the document's paragraphs.
FirstRefPageInDoc	<i>RefPage</i>	Denotes the first reference page in the document.
FirstRubiInDoc	<i>Rubi</i>	Specifies the first rubi composite in the list of the document's rubi composites.
FirstRulingFmtInDoc	<i>RulingFmt</i>	Indicates the first ruling format in the list of the document's ruling formats.
FirstSelectedTiInDoc	<i>FMObject</i>	Specifies the first selected text inset [(FO_TiApiClient, FO_TiText, FO_TiTextTable, or FO_TiFlow ID)] in the list of selected text insets in the document.

Property name	Data Type	Description
FirstSelectedGraphicInDoc	FMObject	Denotes the first selected graphic object [FO_Graphic ID] in the list of selected graphic objects in the document.
SelectedTbl	<i>Tbl</i>	Denotes the table [FO_Tbl ID] containing the selected table cells.
FirstTblFmtInDoc	<i>TblFmt</i>	Denotes the first table format [FO_TblFmt ID] in the list of the document's table formats.
FirstTblInDoc	<i>Tbl</i>	Denotes the first table [FO_Tbl ID] in the list of the document's tables.
FirstTiInDoc	FMObject	Denotes the first text inset [FO_TiApiClient, FO_TiText, FO_TiTextTable, or FO_TiFlow ID] in the list of the document's text insets.
FirstVarFmtInDoc	<i>VarFmt</i>	Specifies the first variable format [FO_VarFmt ID] in the list of the document's variable formats.
FirstVarInDoc	<i>Var</i>	Indicates the first variable [FO_Var ID] in the list of the document's variables.
FirstXRefFmtInDoc	<i>XRefFmt</i>	Denotes the first cross-reference format [FO_XRefFmtID] in the list of the document's cross reference formats.
FirstXRefInDoc	<i>XRef</i>	Specifies the first cross-reference [FO_XRef ID] in the list of the document's cross-references.
HiddenPage	<i>HiddenPage</i>	Denotes a hidden page [FO_HiddenPage ID] in the document.
KeyCatalog	<i>BodyPage</i>	Indicates the last body page [FO_BodyPage ID] in the document.
LastBodyPageInDoc	<i>BodyPage</i>	Indicates the last body page [FO_BodyPage ID] in the document.
LastMasterPageInDoc	<i>MasterPage</i>	Indicates the last master page [FO_MasterPage ID] in the document.
LastRefPageInDoc	<i>RefPage</i>	Specifies the last reference page [FO_RefPage ID] in the document.
LeftMasterPage	<i>MasterPage</i>	Denotes the left master page [FO_MasterPage ID] in the document.

Property name	Data Type	Description
MainFlowInDoc	<i>Flow</i>	Specifies the main flow [FO_Flow ID] in the document.
MarkerTypeNames	<i>Strings</i>	Indicates the list of marker type names.
NextOpenDocInSession	<i>Doc</i>	Denotes the next open document [(FO_Doc ID) in the list of open documents in the session.
RightMasterPage	<i>MasterPage</i>	Denotes the right master page [FO_MasterPage ID] in the document.
AcrobatBookmarkDisplayTags	int	Set to 1 if Include Paragraph Tags in Bookmark Text is on (the paragraph tag is added before the paragraph text in each bookmark); 0, otherwise.
Direction	int	Set or get the direction of the document. Possible values: <ul style="list-style-type: none"> <li><i>Constants.FV_DIR_Inherit</i> - Inherit the direction of the parent</li> <li><i>Constants.FV_DIR_LTR</i> - <i>Left-to-right</i></li> <li><i>FV_DIR_RTL</i> - <i>Right-to-left</i></li> </ul>
ResolvedDirection	int	Get the inherited direction of the document. Possible values: <ul style="list-style-type: none"> <li><i>Constants.FV_DIR_LTR</i> - <i>Left-to-right</i></li> <li><i>FV_DIR_RTL</i> - <i>Right-to-left</i></li> </ul>
AsianComposer	bool	Whether the document will use the Asian Composer layout engine.
DocAcrobatColumnArticleThreads	int	Set to 1 if you want separate article threads for each column; 0 if you want separate article threads for each text frame. The <code>DocPDFNoArticleThread</code> property must be set to <code>False</code> , if the <code>DocAcrobatColumnArticleThreads</code> property is set to <code>True</code> .
DocAcrobatDefaultsChanged	int	Set to 1 if the default heuristics for determining the paragraph level are disabled; 0 otherwise.



Property name	Data Type	Description
DocAcrobatElementList	<i>Strings</i>	Denotes the list of the element tags and context labels to include in bookmarks. The DocPDFElementList property applies only to structured FrameMaker documents.
DocAcrobatElements	int	Set to 1 if elements rather than paragraphs are used for bookmarks; set to 0 otherwise. The DocPDFElements property applies only to structured FrameMaker documents.
DocAcrobatNoArticleThreads	int	Set to 1 if you do not want article threads in the resulting PDF; set to 0 otherwise.
GenerateAcrobatInfo	int	Set to 1 if Generate Adobe Acrobat Data is on. To generate PDF data, you must set the other document print properties as follows: <ul style="list-style-type: none"> <li>PrintToFile - True</li> <li><i>PrintThumbnails - False</i></li> <li><i>PrintSeps - False</i></li> <li><i>PrintBlankPages - True</i></li> <li><i>PrintLastSheetFirst - False</i></li> <li><i>PrintNumCopies - 1</i></li> <li><i>PrintOddPages - True</i></li> <li><i>PrintEvenPages - True</i></li> <li><i>PrintScale - 100%</i></li> </ul>
PDFAllNamedDestinations	int	Set to 1 to create named destinations for all paragraphs and elements in the PDF file generated from this document.
PDFBookmark	bool	Set to True if FrameMaker is to generate bookmarks when saving as PDF; set to False otherwise.

Property name	Data Type	Description
PDFBookmarksOpenLevel	int	<p>Denote the level of bookmarks to have expanded when Acrobat opens the generated PDF document. The value can be any integer, or one of the following defined values:</p> <ul style="list-style-type: none"> <li>• Constants.FV_PDFBookmarksOpenDefaultLevel (-1) <i>Constants.FV_PDFBookmarksOpenAllLevels (-2)</i> <i>Constant.FV_PDFBookmarksOpenNonLevel (-3)</i></li> </ul> <p>If you specify an integer greater than the number of levels in the Bookmarks Settings, Constants.FV_PDFBookmarksOpenAllLevels (-1) takes effect.</p>
PDFConvertCMYKtoRGB	int	<p>When 1, the PDFConvertCMYKtoRGB setting corresponds with setting Convert CMYK colors to RGB in the Save As PDF dialog box.</p>
PDFDestsMarked	int	<p>Set to 1 if the document has paragraphs or elements marked via the <code>MarkedForNamedDestination</code> property.</p> <p>One of two things must happen in order for the PDFDestsMarked property to be True:</p> <ol style="list-style-type: none"> <li>1) The document must be created in version 6.0 or later; the document must be opened in version 6.0 or later <i>The PDF FileSize Optimization client must be run over it to mark all paragraphs or elements that are targets of hypertext links.</i></li> </ol> <p>Normally, your client should not set this value.</p>
PDFDistillerAbsent	int	<p>A value of 1 indicates that Acrobat Distiller is not available. Any other value indicates that Acrobat Distiller is available.</p>

Property name	Data Type	Description
PDFDocInfo	<i>Strings</i>	Contains a list of strings expressing values to be set in the PDF Document Info dictionary when you save the book as PDF. Each dictionary entry is expressed as a pair of strings; the first string expresses the field name, and the second string expresses the field value.
PDFEndPage	string	Indicates then last page of the printing page range, in the FrameMaker numbering style.
PDFJobOption	string	Denotes the name of the Distiller Job Option. If the specified name does not exist in the Distiller Job Options list, then the first Distiller Job Option in the list is used.
PDFJobOptionsAbsent	int	A value of 1 indicates that PDF Job Options are not available.
PDFOpenPage	string	Denote the PDF page number, in the FrameMaker numbering style, at which Acrobat opens the generated PDF document.
PDFPageHeight	int	Indicates the page height for the generated PDF.
PDFPageWidth	int	Indicates the page width for the generated PDF.
PDFPrintPageRange	int	Set to 1 to generate PDF for the specified page range; if set to 0, FrameMaker generates PDF for the entire document or book.
PDFRegistrationMarks	int	Specifies the registration marks for the generated PDF. May be one of: <ul style="list-style-type: none"> <li>Constants.FV_PDFRegistrationMarks-None(0)</li> <li><i>Constants.FV_PDFRegistrationMarksWestern (1)</i></li> <li><i>Constants.FV_PDFRegistrationMarksTombo (2)</i></li> </ul>
PDFSeparateFiles	int	Set to 1, if a separate PDF file should be generated for each document in a book. This property can be set for single document, but is ignored in that case.

Property name	Data Type	Description
PDFStartPage	string	Denotes the first page of the printing page range, in the FrameMaker numbering style.
PDFStructure	int	Set to 1 to generate tagged or structured PDF.
PDFZoomFactor	int	When the <code>PDFZoomType</code> property is set to <code>Constants.FV_PDFZoomNone (0)</code> , the <code>PDFZoomFactor</code> property denotes the zoom percentage of the PDF document (metric 25% to 1600%). If the value of this property is negative or zero, <code>Constants.FV_PDFZoomDefault (1)</code> takes effect.
PDFZoomType	int	Indicates the PDF zoom setting with which Acrobat opens the generated PDF document. The value can be one of: <ul style="list-style-type: none"> <li><code>Constants.FV_PDFZoomDefault (1)</code> <i>Constants.FV_PDFZoomPage (2)</i> <i>Constants.FV_PDFZoomWidth (3)</i> <i>Constants.FV_PDFZoomHeight (4)</i> <i>Constants.FV_PDFZoomNone (0)</i></li> </ul> If a different value is specified, <code>Constants.FV_PDFZoomDefault (1)</code> takes effect.
ChapNumComputeMethod	int	Denotes the component document's chapter numbering type. The possible value is one of: <ul style="list-style-type: none"> <li><code>Constants.FV_NUM_CONTINUE (0x01)</code>: Continue the numbering from the previous chapter. <i>Constants.FV_NUM_RESTART (0x02)</i>: Use the value specified for the <i>Chapter-Number</i> property. <i>Constants.FV_NUM_SAME (0x03)</i>: Use the same chapter number as for the previous file. <i>Constants.FV_NUM_READ_FROM_FILE (0x00)</i>: Use the numbering properties from the document associated with this book component.</li> </ul>

Property name	Data Type	Description
ChapterNumber	int	Use the value of the <code>ChapterNumber</code> property as the chapter number, if the <code>ChapNumComputeMethod</code> property is set to <code>Constants.FV_NUM_RESTART (0x02)</code> .
ChapterNumStyle	int	The numbering style. The possible value is one of: <ul style="list-style-type: none"> <li><code>Constants.FV_NUMSTYLE_NUMERIC (0x00)</code>: Arabic.</li> <li><code>Constants.FV_NUMSTYLE_ROMAN_UC (0x01)</code>: Roman, uppercase.</li> <li><code>Constants.FV_NUMSTYLE_ROMAN_LC (0x02)</code>: Roman, lowercase.</li> <li><code>Constants.FV_NUMSTYLE_ALPHA_UC (0x03)</code>: Alphabetic, uppercase.</li> <li><code>Constants.FV_NUMSTYLE_ALPHA_LC (0x04)</code>: Alphabetic, lowercase.</li> <li><code>Constants.FV_NUMSTYLE_KANJI (0x05)</code>: Kanji.</li> <li><code>Constants.FV_NUMSTYLE_ZENKAKU (0x06)</code>: Zenkaku.</li> <li><code>Constants.FV_NUMSTYLE_ZENKAKU_UC (0x07)</code>: Zenkaku, uppercase.</li> <li><code>Constants.FV_NUMSTYLE_ZENKAKU_LC (0x08)</code>: Zenkaku, lowercase.</li> <li><code>Constants.FV_NUMSTYLE_KANJI_KAZU (0x09)</code>: Kazu.</li> <li><code>Constants.FV_NUMSTYLE_DAIJI (0x0a)</code>: Daiji.</li> <li><code>Constants.FV_NUMSTYLE_TEXT (0x0b)</code>: Text.</li> </ul>
ChapterNumText	string	If the <code>ChapNumStyle</code> property is set to <code>Constants.FV_NUMSTYLE_TEXT (0x0b)</code> , use the value of the <code>ChapterNumText</code> property as the chapter number.
Dictionary	<i>Strings</i>	List of words to accept when spell-checking the document.

Property name	Data Type	Description
DocIsModified	int	Set to 1 if the document has been modified. While this property is read-only, you can modify a document without setting this property to True by setting the Untouchable property to True for the document before your client modifies it. Otherwise, set to 0.
DocIsViewOnly	int	Set to 1 if the document is View Only, else set to 0.
DocOpenType	int	Denotes the type of document opened. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_DOC_TYPE_BINARY (0x01): Frame binary document</li> <li>Constants.FV_DOC_TYPE_TEXT (0x02): ASCII text document</li> <li>Constants.FV_DOC_TYPE_MIF (0x03): MIF document</li> <li>Constants.FV_DOC_TYPE_FILTER (0x04) : A filtered document.</li> </ul>
DocSaveType	int	Denotes the type of document saved. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_DOC_TYPE_BINARY (0x01): Frame binary document</li> <li>Constants.FV_DOC_TYPE_TEXT (0x02): ASCII text document</li> <li>Constants.FV_DOC_TYPE_MIF (0x03): MIF document</li> <li>Constants.FV_DOC_TYPE_FILTER (0x04) : A filtered document.</li> </ul>
DontUpdateTextInsets	int	Set to 1 if FrameMaker is not to automatically update text insets when it opens the document.
DontUpdateXRefs	int	Set to 1 if FrameMaker is not to automatically update cross-references when it opens or prints the document.

Property name	Data Type	Description
FormatOverride	int	Specifies whether there are format overrides at the current insertion point. If the insertion point is in a text range that has a character format applied to it, the FormatOverride property is True if (and only if) the text formatting at the insertion point overrides the character format. If the insertion point is in a text range that does not have a character format applied to it, the FormatOverride property is True if (and only if) the paragraph containing the insertion point has formatting that overrides the Paragraph Catalog format.
IsOnScreen	int	Returns 1 if the document is visible on the screen, otherwise returns 0.
LineNumDistance	Metric	Sets the line number display width, that is, the space in which the line numbers are displayed.
LineNumRestart	int	If set, restarts line number display on each page.
LineNumShow	int	If set, enables the line number display.
Name	string	Denotes the filename of the document.
PageNumComputeMethod	int	Denotes the component document's page numbering type. The possible value is one of: <ul style="list-style-type: none"> <li>Constants.FV_NUM_CONTINUE (0x01): Continue the numbering from the previous file.</li> </ul> <i>Constants.FV_NUM_RESTART (0x02): Restart numbering at the value specified by the FirstPageNum property.</i>
PgfNumComputeMethod	int	Denotes the document's paragraph numbering type. The possible value is one of: <ul style="list-style-type: none"> <li>Constants.FV_NUM_CONTINUE (0x01): Continue the numbering from the previous file.</li> </ul> <i>Constants.FV_NUM_RESTART (0x02): Restart the numbering at 1.</i>

Property name	Data Type	Description
SpecifiedKeyCatalog	F_ObjHandleT	Key Catalog specified for using for the document.
StatusLine	string	Indicates the string that appears in the document status bar. <i>NOTE: This property always returns an empty string when queried; it is effectively write-only.</i> If you set the StatusLine property to a string other than an empty string (""), the string will remain in the status bar until you reset it. To reset the StatusLine property and enable FrameMaker to automatically update the status line with normal status information, set it to an empty string ("").
TextSelection	<i>TextRange</i>	Specifies the currently selected text range or insertion point in the document.
Untouchable	int	Set to 0 by default. Setting this to 1 allows your client to modify a document without setting the DocIsModified property to True.
VolNumComputeMethod	int	Denotes the document's volume numbering type. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_NUM_CONTINUE (0x01): Continue the numbering from the previous volume. <i>Constants.FV_NUM_RESTART (0x02): Use the value specified for the VolumeNumber property.</i> <i>Constants.FV_NUM_SAME (0x03): Use the same volume number as for the previous file</i></li> </ul>
VolumeNumber	int	Use the value of the VolumeNumber property as the volume number, if the VolNumComputeMethod property is set to Constants.FV_NUM_RESTART (0x02).



Property name	Data Type	Description
VolumeNumStyle	int	Denotes the volume numbering style. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_NUMSTYLE_NUMERIC (0x00): Arabic</li> <li>Constants.FV_NUMSTYLE_ROMAN_UC (0x01): Roman numerals, uppercase.</li> <li>Constants.FV_NUMSTYLE_ROMAN_LC (0x02): Roman numerals, lowercase.</li> <li>Constants.FV_NUMSTYLE_ALPHA_UC (0x03): Alphabetic, uppercase.</li> <li>Constants.FV_NUMSTYLE_ALPHA_LC (0x04): Alphabetic, lowercase.</li> <li>Constants.FV_NUMSTYLE_KANJI (0x05): Kanji</li> <li>Constants.FV_NUMSTYLE_ZENKAKU (0x06): Zenkaku</li> <li>Constants.FV_NUMSTYLE_ZENKAKU_UC (0x07): Zenkaku, uppercase.</li> <li>Constants.FV_NUMSTYLE_ZENKAKU_LC (0x08): Zenkaku, lowercase.</li> <li>Constants.FV_NUMSTYLE_KANJI_KAZU (0x09): Kazu</li> <li>Constants.FV_NUMSTYLE_DAIJI (0x0a): Daiji</li> <li>Constants.FV_NUMSTYLE_TEXT (0x0b): Text</li> </ul>
VolumeNumText	string	Use the value of the <code>VolumeNumText</code> property, if the <code>VolNumStyle</code> property is set to <code>Constants.FV_NUMSTYLE_TEXT (0x0b)</code> .
AutoChangeBars	int	Returns 1 if Automatic Change Bars are enabled in the document.
ChangeBarColor	<i>Color</i>	Indicates the spot color (FO_Color ID).
ChangeBarDistance	int	Denotes the distance between change bar and text column.

Property name	Data Type	Description
ChangeBarPosition	int	Denotes the position of the change bars. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CB_COL_LEFT (0x00): Left of Column</li> <li>Constants.FV_CB_COL_RIGHT (0x01): Right of Column</li> <li>Constants.FV_CB_COL_NEAREST (0x02): Side nearest to Page Edge</li> <li>Constants.FV_CB_COL_FURTHEST (0x03): Side farthest from Page Edge</li> </ul>
ChangeBarThickness	int	Denotes the width of the change bars.
ShowAll	int	Set to 1 to display all document conditions, else set to 0.
ShowCondIndicators	int	Set to 1 to display all condition indicators (Format Overrides). Else, set to 0.
EqnIntegralSizeLarge	int	Indicates the point size of integral symbol in large equations (2 pt to 400 pt).
EqnIntegralSizeMed	int	Indicates the point size of integral symbol in medium equations (2 pt to 400 pt).
EqnIntegralSizeSmall	int	Indicates the point size of integral symbol in small equations (2 pt to 400 pt).
EqnLevel1SizeLarge	int	Indicates the point size of level 1 expressions in large equations (2 pt to 400 pt).
EqnLevel1SizeMed	int	Indicates the point size of level 1 expressions in medium equations (2 pt to 400 pt).
EqnLevel1SizeSmall	int	Indicates the point size of level 1 expressions in small equations (2 pt to 400 pt).
EqnLevel2SizeLarge	int	Indicates the point size of level 2 expressions in large equations (2 pt to 400 pt).
EqnLevel2SizeMed	int	Indicates the point size of level 2 expressions in medium equations (2 pt to 400 pt).
EqnLevel2SizeSmall	int	Indicates the point size of level 2 expressions in small equations (2 pt to 400 pt).
EqnLevel3SizeLarge	int	Indicates the point size of level 3 expressions in large equations (2 pt to 400 pt).

Property name	Data Type	Description
EqnLevel3SizeMed	int	Indicates the point size of level 3 expressions in medium equations (2 pt to 400 pt).
EqnLevel3SizeSmall	int	Indicates the point size of level 3 expressions in small equations (2 pt to 400 pt).
EqnSigmaSizeLarge	int	Indicates the point size of the sigma symbol in large equations (2 pt to 400 pt).
EqnSigmaSizeMed	int	Indicates the point size of the sigma symbol in medium equations (2 pt to 400 pt).
EqnSigmaSizeSmall	int	Indicates the point size of the sigma symbol in small equations (2 pt to 400 pt).
Functions	string	Denotes the character format tag of the equation font to apply to Math Functions.
Numbers	string	Denotes the character format tag of the equation font to apply to Math Numbers.
Strings	string	Denotes the character format tag of the equation font to apply to Math Strings.
Symbols	string	Denotes the character format tag of the equation font to apply to Math Symbols.
SymbolsList	<i>Strings</i>	List of math symbol fonts used in the Equation Fonts dialog box.
Variables	string	List of math symbol fonts used in the Equation Fonts dialog box.
HypertextDoValidate	bool	Set to 1 if the next hypertext string sent to the <code>HypertextCommandText</code> property is to be validated.
HypertextCommandText	string	Denotes the hypertext command to parse. Setting this value executes the parser. If the <code>HypertextDoValidate</code> property is True, the command is parsed and validated.

Property name	Data Type	Description
HypertextParseErr	int	<p>Contains a non-zero value if there is a parser error.</p> <p>This property is set to one of the following values if there is a parser error:</p> <ul style="list-style-type: none"> <li> <p>Constants.FV_HypertextSyntaxOK (0) - No parse errors.</p> <p><i>Constants.FV_HypertextEmptyCommand (1) - Hypertext string is empty.</i></p> <p><i>Constants.FV_HypertextUnrecognizedCommand (2) - Cannot map the first keyword to an existing Hypertext-ParsedCmdCode value.</i></p> <p><i>Constants.FV_HypertextMissingArguments (3) - One or more arguments required for the command is missing.</i></p> <p><i>Constants.FV_HypertextExtraArguments (4) - More than the required number of arguments for the command; extra arguments were ignored.</i></p> <p><i>Constants.FV_HypertextBadSyntaxPathSpec (10) - File reference expected for this command, but no valid filepath found.</i></p> <p><i>Constants.FV_HypertextUnanchoredPartialPath (11) - File reference is relative to the current document, but the current document has not been saved; file location could not be calculated.</i></p> <p><i>Constants.FV_HypertextHelpDirNotFound (20) - Default help directory either does not exist (help was not installed) or cannot be found.</i></p> <p><i>Constants.FV_HypertextExpectedANumberParam (30) - Command expected a number but got text; check the value of the</i> <i>HypertextParseBadParam parameter.</i></p> </li> </ul>

Property name	Data Type	Description
HypertextValidateErr	int	<p>Set to a non-zero value if the <code>HypertextDoValidate</code> property is set to True and there is a validation error.</p> <p>The <code>HypertextValidateErr</code> property is set to one of the following values if there is a validation error:</p> <ul style="list-style-type: none"> <li> <p><code>Constants.FV_HypertextValid (0)</code> - No validation errors</p> <p><i>Constants.FV_HypertextUsesDefault-Text (200)</i> - Default text was found as an argument; are you sure the default text is what you want?</p> <p><i>Constants.FV_HypertextFileNotRegular (210)</i> - The referenced file could not be found, or is not a regular file; for example, it could be a directory name.</p> <p><i>Constants.FV_HypertextFileNotMaker-Doc (211)</i> - The referenced file is not created by FrameMaker.</p> <p><i>Constants.FV_HypertextCantOpenDest-File (212)</i> - Cannot open the file; perhaps you do not have permission, or the file is locked.</p> <p><i>Constants.FV_HypertextDestinationLinkNotFound (220)</i> - The referenced file is valid, but cannot find the named link within it.</p> <p><i>Constants.FV_HypertextPageNameNotFound (230)</i> - The referenced file is valid, but the specified page is not found.</p> <p><i>Constants.FV_HypertextUnrecognized-ObjectType (240)</i> - The referenced file is valid, but the link is to an object with an unrecognized object type.</p> <p><i>Constants.FV_HypertextObjectIDNotFound (241)</i> - The file is a link to an object, but cannot find the linked object.</p> <p><i>Constants.FV_HypertextBadMatrixSize (250)</i> - One or both of the matrix dimensions are illegal; The values must be between 1 and 99.</p> </li> </ul>

Property name	Data Type	Description
		<ul style="list-style-type: none"> <li> <p><i>Constants.FV_HypertextMatrixCommandInvalid (251)</i>- One of the commands in the reference page flow for a matrix command has a parse or validation error.</p> <p><i>Constants.FV_HypertextFlowMissingLines (252)</i> - The reference flow for a matrix or popup command is missing one or more lines.</p> <p><i>Constants.FV_HypertextNoNamedFlow (260)</i> - Cannot find the named reference flow for a matrix or popup command.</p> <p><i>Constants.FV_HypertextRecursiveFlow (261)</i> - The reference flow for a matrix or popup command contains nested popup or matrix commands that name a parent reference flow.</p> <p><i>Constants.FV_HypertextMissingPopupMarker (270)</i> - At least one entry in the popup command's reference flow has no hypertext marker in it.</p> <p><i>Constants.FV_HypertextMissingPopupLabelItem (271)</i> - One entry in the popup command's reference flow has no text in it.</p> <p><i>Constants.FV_HypertextEmptyLineInMiddleOfPopup (272)</i> - One entry in the popup command's reference flow has no text in it.</p> <p><i>Constants.FV_HypertextCommandIllegalWithinPopup (273)</i> - Invalid command in the popup command's reference flow; for example, matrix or newlink.</p> <p><i>Constants.FV_HypertextFcodeInvalid (280)</i>- Invalid <i>FCode</i> in the hypertext command.</p> </li> </ul>
HypertextParseBadParam	int	Denotes the index into the string list contained in the HypertextParsedArgs property, in case of a parser error.

---

Property name	Data Type	Description
HypertextParseErrMsg	<code>string</code>	Denotes the message that FrameMaker generates for a parser error.

Property name	Data Type	Description
HypertextParsedCmdCode	int	<p>Denotes the FrameMaker hypertext command in the HypertextCommandText property, as determined by the parser. The possible value for the HypertextParsedCmdCode property is one of:</p> <ul style="list-style-type: none"> <li>• <i>Constants.FV_CmdError (1) -Parser is in an error state</i></li> <li><i>Constants.FV_CmdUnknown (2) - Unknown command</i></li> <li><i>Constants.FV_CmdNoop (3) - Command causes no event.</i></li> <li><i>Constants.FV_CmdAlert (8) - alert command</i></li> <li><i>Constants.FV_CmdAlertTitle (9) - alerttitle command</i></li> <li><i>Constants.FV_CmdExit (10) - exit command</i></li> <li><i>Constants.FV_CmdGoToLink (11) - gotolink command</i></li> <li><i>Constants.FV_CmdGoToLinkFitWin (12) - gotolinkfitwin command</i></li> <li><i>Constants.FV_CmdGoToNew (13) - gotonew command</i></li> <li><i>Constants.FV_CmdGoToPage (14) - gotopage command</i></li> <li><i>Constants.FV_CmdGoToObjectId (15) - gotoObjectId command</i></li> <li><i>Constants.FV_CmdGoToObjectIdFitWin (16) - gotoObjectIdfitwin command</i></li> <li><i>Constants.FV_CmdMatrix (17) - matrix command</i></li> <li><i>Constants.FV_CmdMessage (18) - message command</i></li> <li><i>Constants.FV_CmdNewLink (19) - newlink command</i></li> <li><i>Constants.FV_CmdNextPage (20) - nextpage command</i></li> <li><i>Constants.FV_CmdPreviousPage (21) - previouspage command</i></li> </ul>



Property name	Data Type	Description
		<ul style="list-style-type: none"> <li>           Constants.FV_CmdOpenLink (22) - <i>openlink</i> command            Constants.FV_CmdOpenLinkFitWin (23) - <i>openlinkfitwin</i> command            Constants.FV_CmdOpenNew (24) - <i>opennew</i> command            Constants.FV_CmdOpenObjectId (25) - <i>openObjectId</i> command            Constants.FV_CmdOpenObjectIdFitWin (26) - <i>openObjectIdfitwin</i> command            Constants.FV_CmdOpenPage (27) - <i>openpage</i> command            Constants.FV_CmdPopup (28) - <i>popup</i> command            Constants.FV_CmdPreviousLink (29) - <i>previouslink</i> command            Constants.FV_CmdPreviousLinkFitWin (30) - <i>previouslinkfitwin</i> command            Constants.FV_CmdQuit (31) - <i>quit</i> command            Constants.FV_CmdQuitAll (32) - <i>quitall</i> command         </li> </ul>

Property name	Data Type	Description
HypertextParsedCmdDest	int	<p>Denotes the destination type for link commands in the HypertextCommandText property, as determined by the parser. The possible value for the HypertextParsedCmdDest property is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_DestNowhere (0) - No destination found</li> <li><i>Constants.FV_DestMarkerNewLink (1) - Destination is a newlink</i></li> <li><i>Constants.FV_DestFirstPage (2) - Destination is the first page of a file</i></li> <li><i>Constants.FV_DestLastPage (3) - Destination is the last page of a file</i></li> <li><i>Constants.FV_DestPageNum (4) - Destination is a named page (usually a page number)</i></li> <li><i>Constants.FV_DestFluidFlow (5) - Destination is to a fluid flow document</i></li> <li><i>Constants.FV_DestMarker (6) - Destination is a marker</i></li> <li><i>Constants.FV_DestObjectId (7) - Destination is an object ID (usually for generated hypertext commands)</i></li> <li><i>Constants.FV_DestXRef (8) - Destination is a cross-reference</i></li> </ul>

Property name	Data Type	Description
HypertextParsedCmdDestObjType	int	Denotes the type of the object in the target document, for links to objects. The possible value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ObjectUnknown (0) - Unknown or invalid object</li> <li>Constants.FV_ObjectMarker (1) - Object is a marker</li> <li>Constants.FV_ObjectPgf (2) - Object is a paragraph</li> <li>Constants.FV_ObjectXref (3) - Object is a cross-reference</li> <li>Constants.FV_ObjectGraphic (4) - Object is a graphic</li> <li>Constants.FV_ObjectElement (5) - Object is an element</li> <li>Constants.FV_ObjectTextInset (6) - Object is a text inset</li> <li>Constants.FV_ObjectDataLink (7) - Object is subscribed data</li> </ul>
HypertextParsedCmdDestObjID	int	Denotes the object in the target document for links to objects.
HypertextParsedCmdMatrixRows	int	Denotes the number of rows in the matrix, if the value of the HypertextParsedCmdCode property is set to Constants.FV_CmdMatrix (17)
HypertextParsedCmdMatrixColumns	int	Denotes the number of columns in the matrix, if the value of the HypertextParsedCmdCode property is set to Constants.FV_CmdMatrix (17)
HypertextParsedLinkName	string	Indicates either the value of a newlink command, or a keyword such as FirstPage or LastPage, for links to new targets.
HypertextParsedPageName	string	Denotes the pagenumber, for links to pages.
HypertextParsedFlowName	string	Denotes the name of the flow (on a reference page) that contains the popup or matrix list of commands, for popup and matrix commands.

Property name	Data Type	Description
HypertextParsedClientName	string	Indicates the name of the API client to receive the message, for message commands.
HypertextParsedTitle	string	Indicates the specified title for the alert box, if the value of the <code>HypertextParsedCmdCode</code> property is set to <code>Constants.FV_CmdAlertTitle (9)</code>
HypertextParsedMessage	string	Denotes the specified message for the hypertext command, if the value of the <code>HypertextParsedCmdCode</code> command is set to <code>Constants.FV_CmdAlert (8)</code> , <code>Constants.FV_CmdAlertTitle (9)</code> or <code>Constants.FV_CmdMessage (18)</code> .
HypertextParsedDIFileName	string	Denotes the absolute path to the target file, expressed in platform independent syntax, for links to external files.
MenuBar	<i>Menu</i>	Denotes the identifier of the document's menu bar (FO_Menu ID).
ViewOnlyMenuBar	<i>Menu</i>	Denotes the identifier of the document's menu bar when the document is locked (FO_Menu ID).
FnCustNumString	string	Specifies the characters for custom document footnote numbers.
FnFirstNum	int	Indicates the first document footnote number.
FnFmt	string	Indicates the footnote paragraph tag.
FnHeightPerCol	int	Denotes the maximum height allowed for document footnotes. The permissible range of values is 36 points to 32767 points.
FnInstancePosition	int	Specifies the placement of the footnote number in the footnote. The value is one of: <ul style="list-style-type: none"> <li><code>Constants.FV_FN_POS_SUPER (0x00)</code> - Superscript</li> <li><code>Constants.FV_FN_POS_BASELINE (0x01)</code> - Baseline</li> <li><code>Constants.FV_FN_POS_SUB (0x02)</code> - Subscript</li> </ul>

Property name	Data Type	Description
FnInstancePrefix	string	Indicates the prefix to appear before the document footnote number in the footnote.
FnInstanceSuffix	string	Indicates the suffix to appear after the document footnote number in the footnote.
FnNumComputeMethod	int	The document's footnote numbering type. The value is one of: <ul style="list-style-type: none"><li>• Constants.FV_NUM_CONTINUE (0x01) - Continue the numbering from the previous file. <i>Constants.FV_NUM_RESTART (0x02) - Restart numbering at the value specified by the associated FO_Doc object's FnFirstNum property.</i> <i>Constants.FV_NUM_PERPAGE (0x03) - Restart numbering on each page.</i></li></ul>

Property name	Data Type	Description
FnNumStyle	int	<p>Indicates the document footnote numbering style. The value is one of:</p> <ul style="list-style-type: none"> <li>           Constants.FV_FN_NUM_NUMERIC (0x00) - Arabic  <i>Constants.FV_FN_NUM_ROMAN_UC (0x01) - Roman uppercase</i>  <i>Constants.FV_FN_NUM_ROMAN_LC (0x02) - Roman lowercase</i>  <i>Constants.FV_FN_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</i>  <i>Constants.FV_FN_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</i>  <i>Constants.FV_FN_NUM_KANJI (0x06) - Kanji characters</i>  <i>Constants.FV_FN_NUM_ZENKAKU (0x07) - Zenkaku</i>  <i>Constants.FV_FN_NUM_ZENKAKU_UC (0x08) - Zenkaku uppercase</i>  <i>Constants.FV_FN_NUM_ZENKAKU_LC (0x09) - Zenkaku lowercase</i>  <i>Constants.FV_FN_NUM_KANJI_KAZU (0x0A) - Kazu</i>  <i>Constants.FV_FN_NUM_DAIJI (0x0B) - Daiji</i>  <i>Constants.FV_FN_NUM_CUSTOM (0x05) - Custom numbering</i> </li> </ul>
FnRefPosition	int	<p>Specifies the placement of the footnote reference in the footnote. The value is one of:</p> <ul style="list-style-type: none"> <li>           Constants.FV_FN_POS_SUPER (0x00) - Superscript  <i>Constants.FV_FN_POS_BASELINE (0x01) - Baseline</i>  <i>Constants.FV_FN_POS_SUB (0x02) - Subscript</i> </li> </ul>
FnRefPrefix	string	<p>Indicates the prefix to appear before the number in the document text.</p>

Property name	Data Type	Description
FnRefSuffix	string	Indicates the suffix to appear after the number in the document text.
BottomMargin	int	Indicates the bottom page margin.
ColGap	int	Denotes the size of the gap between text columns
DocIsDoubleSided	int	Set to 1 if the layout of the document is a 2 sided page; 0 otherwise.
FirstPageNum	int	Indicates the page number of the first page.
FirstPageVerso	int	Set to 1 for right first page; 0 for left first page.
LeftMargin	int	Denotes the left page margin.
NumCols	int	Indicates the number of columns on the page.
PageHeight	int	Specifies the height of the document's pages. <i>NOTE: Setting this property, automatically sets the page height for all the body pages of the document.</i>

Property name	Data Type	Description
PageNumStyle	int	<p>Denotes the page numbering style. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_PAGE_NUM_NUMERIC (0x01) - Arabic</li> <li>Constants.FV_PAGE_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>Constants.FV_PAGE_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI (0x05) - Kanji characters</li> <li>Constants.FV_PAGE_NUM_ZENKAKU (0x06) - Zenkaku</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_UC (0x07) - Zenkaku uppercase</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_LC (0x08) - Zenkaku lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI_KAZU (0x09) - Kazu</li> <li>Constants.FV_PAGE_NUM_DAIJI (0x0a) - Daiji</li> </ul>
PageRounding	int	<p>Indicates how to round pages. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_PR_DEL_EMPTY (0x01) - Delete Empty Pages</li> <li>Constants.FV_PR_KEEP_NUM_EVEN (0x02) - Make Page Count Even</li> <li>Constants.FV_PR_KEEP_NUM_ODD (0x03) - Make Page Count Odd</li> <li>FV_PR_DONT_CHANGE (0x04) - Do Not Change Page Count</li> </ul>
PageWidth	int	<p>Specifies the width of the document's pages. <b>NOTE:</b> Setting this property, automatically sets the page width for all the body pages of the document.</p>



Property name	Data Type	Description
PointPageNumStyle	int	Denotes the point page numbering style. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PAGE_NUM_NUMERIC (0x01) - Arabic</li> <li>Constants.FV_PAGE_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>Constants.FV_PAGE_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI (0x05) - Kanji characters</li> <li>Constants.FV_PAGE_NUM_ZENKAKU (0x06) - Zenkaku</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_UC (0x07) - Zenkaku uppercase</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_LC (0x08) - Zenkaku lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI_KAZU (0x09) - Kazu</li> <li>Constants.FV_PAGE_NUM_DAIJI (0x0a) - Daiji</li> </ul>
RightMargin	int	Denotes the Right page margin.
SmartQuotes	int	Returns 1 if Smart Quotes is enabled; returns 0 otherwise.
SmartSpaces	int	Returns 1 if Smart Spaces is enabled; returns 0 otherwise.
TopMargin	int	Denotes the Top Page Margin.
PrintBlankPages	int	Returns 1 if the PageRounding property allows an empty page at end of a document. Returns 0, otherwise.
PrintCollated	int	Returns 1 if Collate is enabled. Returns 0, otherwise.

Property name	Data Type	Description
PrintCols	int	Indicates the number of columns to print, if the <code>PrintThumbnails</code> property is set to <code>True</code> .
PrintEmulsion	int	Specifies the direction of the print emulsion. The value is one of: <ul style="list-style-type: none"> <li>• <code>Constants.FV_EMUL_UP (0)</code> - Emulsion side up</li> <li>• <code>Constants.FV_EMUL_DOWN (1)</code> - Emulsion side down</li> </ul>
PrintEndPage	int	Specifies the number of the last page to print. <b>NOTE:</b> The value of the <code>DocFluidFlow</code> property must be 0, when you use the <code>PrintEndPage</code> property. You cannot print a range of pages when a document is in fluid view.
PrintEndPageName	int	Denotes the page number string for the last page to be printed. Use this property when the pages are numbered with a style other than <code>Constants.FV_PAGE_NUM_NUMERIC (0x01)</code> . <b>NOTE:</b> The value of the <code>DocFluidFlow</code> property must be 0, when you use the <code>PrintEndPageName</code> property. You cannot print a range of pages when a document is in fluid view.
PrintEndPoint	int	Denotes the number of the last point page to print.
PrinterName	string	Specifies the name of the printer. <b>NOTE:</b> Setting the <code>PrinterName</code> property on Windows has no effect. <b>NOTE:</b> You can set the printer to the default printer by specifying the <code>PrinterName</code> as null.
PrintEvenPages	int	Set to 1 if Print Even-Numbered Pages is enabled. Set to 0, otherwise.
PrintFileName	string	Name of the file to which to print. You can set the filename to the default filename by specifying the value of the <code>PrintFileName</code> property as null.

Property name	Data Type	Description
PrintImaging	int	Specifies the type of print imaging. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_IMG_POSITIVE (0)</li> <li>Constants.FV_IMG_NEGATIVE (1)</li> </ul>
PrintLastSheetFirst	int	Set to 1 to print the last sheet first. Else, set to 0.
PrintLowRes	int	Set to 1 to print in low resolution. Else, set to 0.
PrintManualFeed	int	Set to 1 if Manual feed is enabled. Else, set to 0.
PrintNumCopies	int	Denotes the number of copies to print.
PrintOddPages	int	Set to 1 if Print Odd-Numbered Pages is enabled. Else, set to 0.
PrintPaperHeight	int	Specifies the height of the printing paper.
PrintPaperWidth	int	Indicates the width of the printing paper.
PrintRegistrationMarks	int	Set to 1 if Registration Marks is enabled; Set to 0 otherwise.
PrintRows	int	Indicates the number of rows to print if the PrintThumbnails property is set to True.
PrintScale	int	Denotes the Print scale factor expressed as a percentage metric (0% to 100%).For print scale percentages, the value <<16 or 0x10000 specifies 100%.
PrintScope	int	Denotes the pages to print. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PR_ALL (1) - Print all pages</li> <li>Constants.FV_PR_RANGE (2) - Print a range of pages</li> </ul> <p><b>NOTE:</b> The value of the DocFluidFlow property must be set to 0 when you set the PrintScope property. You cannot print a range of pages when a document is in the fluid view.</p>
PrintSepts	int	Returns 1 if Print Separations is enabled. Else, returns 0.

Property name	Data Type	Description
PrintStartPage	int	Denotes the number of the first page to print. <i>NOTE: The value of the DocFluidFlow property must be set to 0 when you set the PrintStartPage property. You cannot print a range of pages when a document is in the fluid view.</i>
PrintStartPageName	int	Indicates the page number string for the first page to print. This property is valid only when the pages are numbered with a style other than Constants.FV_PAGE_NUM_NUMERIC (0x00) <i>NOTE: The value of the DocFluidFlow property must be set to 0 when you set the PrintStartPage property. You cannot print a range of pages when a document is in the fluid view.</i>
PrintStartPoint	int	Indicates the number of the first point page to print.
PrintThumbnails	int	Returns 1 if Print Thumbnails is enabled. Returns 0, otherwise.
PrintToFile	int	Returns 1 if Print Only to File is enabled. Else, returns 0.
SkipBlankSeps	int	Returns 1 if Skip Blank Separations (do not print blank color separations) is enabled. Else, returns 0.
TrapwiseCompatibility	bool	Set to 1 if Trapwise Compatibility is enabled. Else, set to 0. <i>NOTE: Setting the TrapwiseCompatibility parameter to True automatically sets the PrintToFile parameter to True and the PrintSeps parameter to False.</i>
NarrowRubiSpaceForOther	int	Denotes the value for a composite ruby object. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_Wide (0)</li> <li>Constants.FV_Narrow (1)</li> <li>Constants.FV_Proportional (2)</li> </ul>

Property name	Data Type	Description
RubiOverhang	int	Set to 1 if Rubi is allowed to overhang, else set to 0.
RubiSize	int	Specifies the scaling factor for rubi text expressed as percentage of the current font size (metric 1% to 1000%). <i>NOTE: If the RubiSize property and the RubiFixedSize property are both set, the most recently set property value is used.</i>
RubiFixedSize	int	Specifies the fixed size for all rubi text (metric 2 points to 400 points). <i>NOTE: If the RubiFixedSize property and the RubiSize property are both set, the most recently set property value is used.</i>
WideRubiSpaceForOther	int	Denotes the value for a composite ruby object. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_Wide (0)</li> <li>Constants.FV_Narrow (1)</li> <li>Constants.FV_Proportional (2)</li> </ul>
ElementSelection	<i>ElementRange</i>	Denotes the currently selected element range in the document.
CustomElementList	<i>Strings</i>	Contains the list of tags to display when the ElementCatalogDisplay property is set to Constants.FV_ELCAT_CUSTOM (0x04).
DefaultExclusions	<i>Strings</i>	Indicates the list of exclusions inherited when a document is included in a structured book.
DefaultInclusions	<i>Strings</i>	Indicates the list of inclusions inherited when a document is included in a structured book.
ElementBoundaryDisplay	int	Specifies the Element Boundary Display options. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ELEM_DISP_NONE (0) - Do not display any element boundaries</li> <li>Constants.FV_ELEM_DISP_BRACKETS (1) - Display the bracketed boundaries</li> <li>Constants.FV_ELEM_DISP_TAGS (2) - Display the element tags</li> </ul>

Property name	Data Type	Description
ElementCatalog	<i>ElementCatalogEntries</i>	Displays the list of elements in the Element Catalog.
ElementCatalogDisplay	int	Indicates the catalog display options. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ELCAT_STRICT (0x00)- Valid children for working from start to finish</li> <li>Constants.FV_ELCAT_LOOSE (0x01) - Valid children for working in any order</li> <li>Constants.FV_ELCAT_CHILDREN (0x02) - Children allowed anywhere in the parent</li> <li>Constants.FV_ELCAT_ALL (0x03) - All elements</li> <li>Constants.FV_ELCAT_CUSTOM (0x04) - The list of tags specified by the CustomElementList property.</li> </ul>
FirstElementDefInDoc	<i>ElementDef</i>	Denotes the first element definition in the list of element definitions in the document (FO_ElementDef ID).
FirstFmtChangeListInDoc	<i>FmtChangeList</i>	Specifies the ID of the first format change list in the list of format change lists in the book.(FO_FmtChangeList ID).
MaxBottomMargin	int	Specifies the maximum bottom margin allowed in the document.
MaxFirstIndent	int	Specifies the maximum first indent allowed in the document.
MaxFontSize	int	Denotes the maximum font size allowed in the document.
MaxLeading	int	Denotes the maximum leading allowed in the document.
MaxLeftIndent	int	Specifies the maximum left indent allowed in the document.
MaxLeftMargin	int	Specifies the maximum left margin allowed in the document.

Property name	Data Type	Description
MaxRightIndent	int	Specifies the maximum right indent allowed in the document.
MaxRightMargin	int	Specifies the maximum right margin allowed in the document.
MaxSpaceAbove	int	Denotes the maximum space above paragraph allowed in the document.
MaxSpaceBelow	int	Denotes the maximum space below paragraph allowed in the document.
MaxStretch	int	Denotes the maximum character stretch (set width) expressed as a percentage of normal stretch for the font (metric -10% to 1000%).
MaxTabPosition	int	Specifies the maximum tab position allowed in the document.
MaxTopMargin	int	Specifies the maximum top margin allowed in the document.
MinBottomMargin	int	Specifies the minimum bottom margin allowed in the document.
MinFirstIndent	int	Specifies the minimum first indent allowed in the document.
MinFontSize	int	Denotes the minimum font size allowed in the document.
MinLeading	int	Denotes the minimum leading allowed in the document.
MinLeftIndent	int	Denotes the minimum left indent allowed in the document.
MinLeftMargin	int	Denotes the minimum left margin allowed in the document.
MinRightIndent	int	Denotes the minimum right indent allowed in the document.
MinRightMargin	int	Denotes the minimum right margin allowed in the document.
MinSpaceAbove	int	Denotes the minimum space above the paragraph allowed in the document.

Property name	Data Type	Description
MinSpaceBelow	int	Denotes the minimum space below the paragraph allowed in the document.
MinStretch	int	Indicates the minimum character stretch (set width) expressed as a percentage of normal stretch for the font (metric -10% to 1000%)
MinTabPosition	int	Indicates the minimum tab position allowed in the document.
MinTopMargin	int	Indicates the minimum top margin allowed in the document.
NewElemAttrDisplay	int	Specifies the attribute display properties for new elements. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ATTR_DISP_NONE (1) - Do not display attributes</li> <li>Constants.FV_ATTR_DISP_REQSPEC (2) - Display required and specified attributes</li> <li>Constants.FV_ATTR_DISP_ALL (3) - Display all attributes</li> </ul>
NewElemAttrEditing	int	Specifies when the Edit Attributes dialog box appears for new elements. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ATTR_EDIT_NONE (0)</li> <li>Constants.FV_ATTR_EDIT_REQUIRED (1)</li> <li>Constants.FV_ATTR_EDIT_ALWAYS (2)</li> </ul>
SeparateInclusions	int	Returns 1 if inclusions are listed separately in the Element Catalog. Else, returns 0.
StructuredApplication	string	The name of the structure application that is associated with the document. If the document has no associated structure application, the <code>GetString()</code> method returns an empty string.
UseInitialStructure	int	Returns 1 if FrameMaker inserts the initial structure for new elements. Else, returns 0.
XmlDocType	string	Contains the DOCTYPE parameter. from the source XML.



Property name	Data Type	Description
XmlEncoding	string	<p>Contains the encoding parameter of the XML Declaration for the source XML document. The string is empty if no encoding is specified.</p> <p>If the <code>XmlEncoding</code> property is set, the XML Declaration contains the encoding parameter with the set value on <code>Save As XML</code> operation.</p>
XmlFileEncoding	string	<p>Specifies the encoding that is detected for the source XML book. If no encoding is specified for the source XML, the value of the <code>XmlEncoding</code> property will be an empty string. In that case, if this string is set, it determines the encoding to use when saving as XML.</p> <p>If the <code>XmlEncoding</code> property has a value, the value of the <code>XmlFileEncoding</code> property may be null.</p>
XmlPublicId	string	<p>Indicates the DOCTYPE public identifier for the source XML document.</p>
XmlStandAlone	int	<p>An integer that specifies the XML standalone parameter for the XML document that is the source of the current document. The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_XML_STANDALONE_YES</code> (1)</li> <li><code>Constants.FV_XML_STANDALONE_NO</code> (2)</li> <li><code>Constants.FV_XML_STANDALONE_NO-DEC</code> (3)</li> <li><code>Constants.FV_XML_STANDALONE_-NONE</code> (4)</li> </ul> <p>The standalone parameter is declared in the XML Declaration.</p> <p>For a file with no XML Declaration, the value is <code>Constants.FV_XML_STANDALONE_NODEC</code> (3).</p> <p>For an XML Declaration with no standalone parameter, this value is <code>Constants.FV_XML_STANDALONE_NONE</code> (4).</p>

Property name	Data Type	Description
XmlStyleSheet	string	<p>Specifies the XML stylesheet processing instruction to write out to XML when saving the document as XML. The ESTK does not verify that you use correct syntax in this string. The value you set should not include the PI delimiters, &lt;? and ?&gt;. For example, the string you supply for my.css may be:</p> <pre> type="text\\css" href="my.css" </pre> <p>Only use this string to set a specific stylesheet specification.</p> <p><b>NOTE:</b> The <i>GetString()</i> method always returns null for this property. To get the list of stylesheet specifications associated with a document, use the <i>XmlStyleSheetList</i> property.</p>
XmlStyleSheetList	<i>Strings</i>	<p>Contains a list of stylesheet processing instructions for the current document. A document can have more than one stylesheet specification associated with it. The ESTK does not verify that you use the correct syntax in these strings. The value you set should not include the PI delimiters, &lt;? and ?&gt;. For example, the string you supply for my.css may be:</p> <pre> type="text\\css" href="my.css" </pre> <p>Setting a list to the <i>XmlStyleSheetList</i> property completely overwrites the preceding list.</p>
XmlSystemId	string	<p>Specifies the DOCTYPE system identifier for the source XML document.</p>
XmlUseBOM	int	<p>Indicates whether a byte order mark was detected when opening the source XML. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_XML_USEBOM_YES (1)</li> <li>Constants.FV_XML_USEBOM_NO (2)</li> </ul> <p>If this value is set to Constants.FV_XML_USEBOM_YES (1), FrameMaker writes a byte order mark in the resulting XML, when saving as XML.</p>

Property name	Data Type	Description
XmlVersion	string	Indicates the XML Version that was specified in the XML Declaration when the file was opened. If no XML version was specified, the GetString() method returns an empty string. If this string contains an invalid XML declaration, a parsing error occurs when the document is saved as XML.
XmlWellFormed	int	Indicates whether the source XML qualified as well formed. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_XML_WELLFORMED_YES (1)</li> <li>Constants.FV_XML_WELLFORMED_NO (2)</li> </ul>
TblFnCellPosition	int	Denotes the placement of footnote number in footnote text. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_FN_POS_SUPER (0x00) - Superscript</li> <li>Constants.FV_FN_POS_BASELINE (0x01) - Baseline</li> <li>Constants.FV_FN_POS_SUB (0x02) - Subscript</li> </ul>
TblFnCellPrefix	string	Specifies the prefix to appear before the table footnote number in table cell.
TblFnCellSuffix	string	Specifies the suffix to appear after the table footnote number in table cell.
TblFnCustNumString	string	Specifies the characters for custom table footnote numbers.
TblFnFmt	string	Represents the paragraph tag of the table footnote.

Property name	Data Type	Description
TbIFnNumStyle	int	<p>Denotes the footnote numbering style for tables in the document. The value is one of:</p> <ul style="list-style-type: none"> <li>           Constants.FV_FN_NUM_NUMERIC (0x00) - Arabic  <i>Constants.FV_FN_NUM_ROMAN_UC (0x01) - Roman uppercase</i>  <i>Constants.FV_FN_NUM_ROMAN_LC (0x02) - Roman lowercase</i>  <i>Constants.FV_FN_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</i>  <i>Constants.FV_FN_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</i>  <i>Constants.FV_FN_NUM_KANJI (0x06) - Kanji characters</i>  <i>Constants.FV_FN_NUM_ZENKAKU (0x07) - Zenkaku</i>  <i>Constants.FV_FN_NUM_ZENKAKU_UC (0x08) - Zenkaku uppercase</i>  <i>Constants.FV_FN_NUM_ZENKAKU_LC (0x09) - Zenkaku lowercase</i>  <i>Constants.FV_FN_NUM_KANJI_KAZU (0x0A) - Kazu</i>  <i>Constants.FV_FN_NUM_DAIJI (0x0B) - Daiji</i>  <i>Constants.FV_FN_NUM_CUSTOM (0x05) - Custom numbering.</i> </li> </ul>
TbIFnPosition	int	<p>Denotes the placement of the footnote number in the text. The value is one of:</p> <ul style="list-style-type: none"> <li>           Constants.FV_FN_POS_SUPER (0x00) - Superscript  <i>Constants.FV_FN_POS_BASELINE (0x01) - Baseline</i>  <i>Constants.FV_FN_POS_SUB (0x02) - Subscript</i> </li> </ul>
TbIFnPrefix	string	Indicates the prefix to appear before the number in the table footnote.
TbIFnSuffix	string	Indicates the suffix to appear after the number in the table footnote.

Property name	Data Type	Description
Capitalization	int	Indicates the type of capitalization. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAPITAL_CASE_NORM (0)</li> <li>Constants.FV_CAPITAL_CASE_SMALL (1)</li> <li>Constants.FV_CAPITAL_CASE_LOWER (2)</li> <li>Constants.FV_CAPITAL_CASE_UPPER (3)</li> </ul>
ChangeBar	int	Returns 1 if Changebars are enabled. Else, returns 0.
CharTag	string	Indicates the name of the character format tag.
Color	<i>Color</i>	Denotes the Spot color (FO_Color ID).
CondFmtIsShown	int	Returns 1 if the document conditions are shown. Else, returns 0.
CombinedFontFamily	<i>CombinedFontDefn</i>	Contains the Combined font definition (FO_CombinedFontDefn)
FontEncodingName	string	Specifies the font encoding.
FontAngle	int	Denotes the font angle (specifies an index into the array of font angles provided by the session property, <i>FontAngleNames</i> ).
FontFamily	int	Denotes the font family (specifies an index into the array of font families provided by the session property, <i>FontFamilyNames</i> ).
FontPlatformName	string	Indicates the name that uniquely identifies a font on a specific platform.
FontPostScriptName	string	Indicates the name given to a font when it is sent to a PostScript printer.
FontSize	int	Specifies the font size (2 point to 400 point).
FontVariation	int	Denotes the font variation (specifies an index into the array of font variations provided by the session property <i>FontVariationNames</i> ).

Property name	Data Type	Description
FontWeight	int	Denotes the font weight (specifies an index into the array of font weights provided by the session property <code>FontWeightNames</code> ).
InCond	Objects	Indicates the condition tags that apply to the text (an array of <code>FO_CondFmt</code> IDs).
KernX	int	Specifies the horizontal kern value for manual kerning expressed as a percentage of an em (metric – 100% to 1000%). A positive value moves a character right and a negative value moves a character left.
KernY	int	Specifies the vertical kern value for manual kerning expressed as a percentage of an em (metric – 100% to 1000%). A positive value moves characters up and a negative value moves characters down.
Overline	int	Returns 1 if <code>Overline</code> style is enabled, else returns 0.
PairKern	int	Returns 1 if <code>Pair Kern</code> is enabled, else returns 0.
Position	int	Denotes the text position relative to the baseline of text. The value is one of: <ul style="list-style-type: none"> <li><code>Constants.FV_POS_NORM (0)</code> - Normal</li> <li><code>Constants.FV_POS_SUPER (1)</code> - <i>Superscript</i></li> <li><code>Constants.FV_POS_SUB (2)</code> - <i>Subscript</i></li> </ul>
SepOverride	<i>Color</i>	Specifies the custom color separation override ( <code>FO_Color</code> ID).
Stretch	int	Denotes the character stretch (set width) expressed as a percentage of normal stretch for the font (metric –10% to 1000%).
Strikethrough	int	Returns 1 if <code>Strikethrough</code> style is enabled. Else, returns 0.

Property name	Data Type	Description
StyleOverrides	int	Specifies the style condition indicators for conditional text. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CN_NO_OVERRIDE (0)</li> <li>Constants.FV_CN_OVERLINE (1)</li> <li>Constants.FV_CN_STRIKETHROUGH (2)</li> <li>Constants.FV_CN_SINGLE_UNDERLINE (3)</li> <li>Constants.FV_CN_DOUBLE_UNDERLINE (4)</li> </ul>
Underlining	int	Denotes the style of underlining. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CB_NO_UNDERLINE (0)</li> <li>Constants.FV_CB_SINGLE_UNDERLINE (1)</li> <li>Constants.FV_CB_DOUBLE_UNDERLINE (2)</li> <li>Constants.FV_CB_NUMERIC_UNDERLINE (3)</li> </ul>
UseSepOverride	int	Returns 1 if the value of the <code>SepOverride</code> property overrides the default color separation. Else, returns 0.
LineBreakAfter	string	Indicates the characters at which it is permissible to break lines.
SmallCapsSize	int	Indicates the scaling factor for small caps expressed as a percentage of current font size (metric 1% to 1000%).
SmallCapsStretch	int	Denotes the character stretch (set width) for small caps expressed as a percentage of normal stretch for the font (metric -10% to 1000%).
SubScriptShift	int	Denotes the baseline offset of subscripts expressed as a percentage of current font size (metric 1% to 1000%).
SubScriptSize	int	Indicates the scaling factor for subscripts expressed as a percentage of current font size (metric 1% to 1000%).

Property name	Data Type	Description
SubScriptStretch	int	Indicates the character stretch (set width) for subscripts expressed as a percentage of normal stretch for the font (metric -10% to 1000%).
SuperScriptShift	int	Denotes the baseline offset of superscripts expressed as percentage of current font size (metric 1% to 1000%).
SuperScriptSize	int	Specifies the scaling factor for superscripts expressed as percentage of the current font size (metric 1% to 1000%)
SuperScriptStretch	int	Specifies the character stretch (set width) for superscripts expressed as a percentage of normal stretch for the font (metric -10% to 1000%).
IsIconified	int	Returns 1 if the document window is iconified. Else, returns 0.
IsInFront	int	Set to 1 if the document window is in front of other windows in the FrameMaker product session. Else, set to 0.
Label	string	Denotes the title in the document window title bar.
ScreenHeight	int	Denotes the height of the document window in pixels.
ScreenWidth	int	Denotes the width of the document window in pixels.
ScreenX	int	Indicates the offset of the document window in pixels from the left side of the screen (or the left of the FrameMaker application window). If you set a value that results in the document window being off the screen, that value is ignored and the old value is retained.
ScreenY	int	Indicates the offset of the document window in pixels from the top of the screen (or the top of the FrameMaker application window). If you set a value that results in the document window being off the screen, that value is ignored and the old value is retained.



Property name	Data Type	Description
SnapAngle	int	Specifies the angle of rotation for Snap Rotate.
SnapGridUnits	int	Denotes the units for Snap Grid spacing. The permissible range of values is from 0 to 32768 points.
SpotColorView	int	Denotes the spot color separation view. The values range from 0 to 6.0 specifies View 1, 1 specifies View 2, and so on.
ViewBorders	int	Returns 1 if Borders is enabled, else returns 0.
ViewDisplayUnits	int	Denotes the MetricT equivalent of one unit in the current Display Units. For example, if Display Units is points, this parameter returns 65536.
ViewFontSizeUnits	int	Denotes the MetricT equivalent of one unit in the current Font Size Unit. Font size units can be either Points or Q. If Points, this returns 65536. If Q, this returns 47098.
ViewGrid	int	Returns 1 if View Grid is enabled. Else, returns 0.
ViewGridUnits	int	Denotes the units for Grid Lines.
ViewNoGraphics	int	Returns 1 if Graphics is not enabled. Else, set to 0.
ViewPageScrolling	int	Denotes the Page scrolling technique. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_SCROLL_VARIABLE (0)</li> <li>Constants.FV_SCROLL_HORIZONTAL (1)</li> <li>Constants.FV_SCROLL_VERTICAL (2)</li> <li>Constants.FV_SCROLL_FACING (3)</li> </ul>
ViewRulers	int	Returns 1 if Rulers is enabled. Else, returns 0.
ViewRulerUnits	int	Denotes the units to display rulers.
ViewTextSymbols	int	Returns 1 if Text Symbols is enabled. Else, returns 0.
Zoom	int	Specifies the zoom percentage of document (metric 25% to 1600%).

Property name	Data Type	Description
DocFluidFlow	<i>Flow</i>	Specifies the flow to set to fluid view. To turn this off, set the value of this property to 0.
ViewOnlyDeadCodes	<i>UInts</i>	Indicates the F-codes that cannot be executed in the document.
ViewOnlySelect	int	Specifies whether user can select text or graphics in the document. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_VOS_USER_ONLY (1) - the user can select text when pressing modifier keys, and link targets (cross-reference sources and newliniks) do not highlight. <i>Constants.FV_VOS_NONE (2) - the user cannot select text, and links targets do not highlight.</i> <i>Constants.FV_VOS_YES(3) - the user can select text (using modifier keys) and link targets are highlighted.</i></li> </ul>
ViewOnlyWinBorders	int	Returns 1 if the document has normal document borders; returns 0 if the document scroll bars and border buttons are suppressed.
ViewOnlyWinMenubar	int	Returns 1 if the document has a document window menu bar. Else, returns 0.
ViewOnlyWinPalette	int	Returns 1 if the document is a palette. Else, returns 0.
ViewOnlyWinPopup	int	Returns 1 if the document window pop-up menu is available. Else, returns 0.

Property name	Data Type	Description
ViewOnlyXRef	int	<p>Specifies the behavior of cross-references in the document. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_VOX_NOT_ACTIVE(0) - Cross references are not active.</li> </ul> <p><i>Constants.FV_VOX_GOTO_BEHAVIOR(1) - Internal cross-references are active.</i></p> <p><i>Constants.FV_VOX_OPEN_BEHAVIOR(2) - External cross-references are active.</i></p> <p><i>Constants.FV_VOX_ALERT(3) - Alert appears when cross-reference is clicked.</i></p>
PDFGenerateForReview	int	<p>This flag is used to generate a PDF for review workflow that is getting review comments in PDF and importing them back to FrameMaker document. It helps in importing back the comments correctly. When this flag is turned ON, the behavior is same as if the "Generate for review only" checkbox is ON in the PDF setup dialog.</p>
PDFViewPDF	int	<p>Set to 1 to open the PDF after creating it with the <code>Save as PDF</code> command.</p>
StructuredApplicationForOpen	string	<p>Specifies the application that has been used to open the document.</p>
SecNumComputeMethod	int	<p>Denotes the type of numbering applied on a particular section of a document. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_NUM_CONTINUE (0x01)</li> <li>Constants.FV_NUM_RESTART (0x02)</li> <li>Constants.FV_NUM_SAME (0x03)</li> </ul>

Property name	Data Type	Description
SectionNumStyle	int	<p>Denotes the section numbering style. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_PAGE_NUM_NUMERIC(0x00) - Arabic</li> <li>Constants.FV_PAGE_NUM_ROMAN_UC(0x01) - Roman uppercase</li> <li>Constants.FV_PAGE_NUM_ROMAN_LC(0x02) - Roman lowercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_UC(0x03) - Alphabetic uppercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_LC(0x04) - Alphabetic lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI(0x05) - Kanji characters</li> <li>Constants.FV_PAGE_NUM_ZENKAKU(0x06) - Zenkaku</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_UC(0x07) - Zenkaku uppercase</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_LC(0x08) - Zenkaku lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI_KAZU(0x09) - Kazu</li> <li>Constants.FV_PAGE_NUM_DAIJI(0x0a) - Daiji</li> </ul>
SectionNumber	int	Denotes the section number in the document.
SectionNumText	string	Indicates the text to use for the SectionNumber parameter.
SubsecNumComputeMethod	int	<p>Denotes the type of numbering applied on a particular subsection of a document. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_NUM_CONTINUE (0x01)</li> <li>Constants.FV_NUM_RESTART (0x02)</li> <li>Constants.FV_NUM_SAME (0x03)</li> </ul>

Property name	Data Type	Description
SubsectionNumStyle	int	Denotes the subsection numbering style. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PAGE_NUM_NUMERIC(0x00) - Arabic</li> <li>Constants.FV_PAGE_NUM_ROMAN_UC(0x01) - Roman uppercase</li> <li>Constants.FV_PAGE_NUM_ROMAN_LC(0x02) - Roman lowercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_UC(0x03) - Alphabetic uppercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_LC(0x04) - Alphabetic lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI(0x05) - Kanji characters</li> <li>Constants.FV_PAGE_NUM_ZENKAKU(0x06) - Zenkaku</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_UC(0x07) - Zenkaku uppercase</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_LC(0x08) - Zenkaku lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI_KAZU(0x09) - Kazu</li> <li>Constants.FV_PAGE_NUM_DAIJI(0x0a) - Daiji</li> </ul>
SubsectionNumber	int	Denotes the subsection number in the document
SubsectionNumText	string	Indicates the text to use for the SubsectionNumber parameter.
TrackChangesOn	bool	Set to True if Track Changes are on in the document. Else, set to False.

Property name	Data Type	Description
PreviewState	int	Sets/retrieves the state of Track Text Edits of the document. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PREVIEW_OFF_TRACK_CHANGE (0) - Displays text edits using the defined styles.</li> <li>Constants.FV_PREVIEW_ON_ORIGINAL (1)- Preview of the document as if no text edits were done.</li> <li>Constants.FV_PREVIEW_ON_FINAL (2)- Preview of the document as if all the text edits were incorporated.</li> </ul>
TrackChangesAddedColor	Color	Sets/retrieves the color in which tracked additions are displayed.
TrackChangesDeletedColor	Color	Sets/retrieves the color in which tracked deletions are displayed.
ReviewerNameList	Strings	Retrieves the list of reviewers of a document.
FirstAttrCondExprInDoc	AttrCondExpr	Denotes the first Attribute Conditional Expression in the list of attribute conditional expressions in the document (FO_AttrCondExpr).
BooleanConditionExpression	string	Denotes the actual conditional expression used in the document.
BooleanConditionState	int	Set to 1 to indicate that the conditional expression defined by the BooleanConditionExpression parameter is applied to the document. Set to 0 to indicate that the condition is not applied to the document.
BooleanConditionExpressionTag	string	Indicates the tag name of the expression applied in the document.
ShowElementDescriptiveNames	bool	Set to True to show element descriptive names in the element catalog, as specified in element definition.

Property name	Data Type	Description
UseInitialStructureOfAutoInsertedElements	bool	If set to True, auto-insertion rules will be processed recursively. For example: If an element is inserted automatically, and if auto-insertion rules exist for this element in the element definition, then those rules are also processed.
UserString	string	A string to which clients can store private data.
DitaMaxRefLevels	int	Indicates the number of references to be opened while opening a DITA file. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_LEVELS_ALL-Open all references within the document.</li> <li>Constants.FV_LEVELS_DEFAULT-Open cross references till the level specified in the ditafm.ini file.</li> </ul>

## Doc methods

*AddNewBuildExpr, AddText, CenterOnText, Clear, ClearAllChangebars, Close, Compare, Copy, Cut, DeleteBuildExpr, DeleteText, DeleteUnusedCharFmt, DeleteUnusedFmts, DeleteUnusedPgfmt, DeleteUnusedTblFmt, DemoteElement, ElementLocToTextLoc, Find, GetActiveBuildExpr, GetBuildExpr, GetBuildExprCatalog, GetNamedAttrCondExpr, GetNamedCharFmt, GetNamedColor, GetNamedCombinedFontDefn, GetNamedCondFmt, GetNamedElementDef, GetNamedFmtChangeList, GetNamedMarkerType, GetNamedMasterPage, GetNamedObject, GetNamedPgfmt, GetNamedRefPage, GetNamedRulingFmt, GetNamedTblFmt, GetNamedUnanchoredFrame, GetNamedVarFmt, GetNamedXRefFmt, GetProps, GetTextForRange, GetTextForRange2, GetTextPropVal, GetTextProps, GetTextVal, GetUniqueObject, HypertextCommand, Import, MergeIntoFirst, MergeIntoLast, NewAnchoredAFrame, NewAnchoredFn, NewAnchoredFormattedObject, NewAnchoredFormattedTbl, NewAnchoredFormattedVar, NewAnchoredFormattedXRef, NewAnchoredMarker, NewAnchoredObject, NewAnchoredTbl, NewAnchoredTiApiClient, NewArc, NewEllipse, NewFlow, NewGraphicObject, NewGroup, NewInset, NewInlineComponentOfType, NewLine, NewMath, NewMathML, NewNamedAttrCondExpr, NewNamedCharFmt, NewNamedColor, NewNamedCombinedFontDefn, NewNamedCondFmt, NewNamedElementDef, NewNamedFmtChangeList, NewNamedMarkerType, NewNamedMasterPage, NewNamedObject, NewNamedPgfmt, NewNamedRefPage, NewNamedRulingFmt, NewNamedTblFmt, NewNamedVarFmt, NewNamedXRefFmt, NewPolygon, NewPolyline, NewRectangle, NewRoundRect, NewSeriesBodyPage, NewSeriesObject, NewSeriesPgfmt, NewTable, NewTextFrame, NewTextLine, NewUnanchoredFrame, ObjectValid, Paste, PromoteElement, QuickSelect, Redisplay, Reformat, Rehyphenate, ResetEqnSettings, ResetReferenceFrames, RestartPgfnNumbering, Save, ScrollToText, SetActiveBuildExpr, SetElementRange, SetProps, SetTextPropVal, SetTextProps, SetTextVal, SilentPrintDoc, SimpleImportElementDefs, SimpleImportFormats, SimpleSave, SplitElement, TextLocToElementLoc,*

[TrackChangesAcceptAll](#), [TrackChangesRejectAll](#), [UnWrapElement](#), [UpdateVariables](#), [UpdateXRef](#), [UpdateXRefs](#).

## Element

Property name	Data Type	Description
AttrDisplay	int	Specifies element's attribute display properties. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ATTR_DISP_NONE (1) - Do not display attributes</li> <li>Constants.FV_ATTR_DISP_REQSPEC (2) - Display required and specified attributes</li> <li>Constants.FV_ATTR_DISP_ALL (3) - Display all attributes</li> </ul>
Attributes	<a href="#">Attributes</a>	The element's attributes.
ContextLabel	string	The context label (if any) that is applied to the element.
ElementIsCollapsed	int	Returns 1 if the element is collapsed in Structure View. Returns 0 otherwise.
ElementIsTopicRef	bool	Determines if the element is a topic reference.
ElementIsTopicHead	bool	Determines if the element is a topic heading.
ElementIsTopicGroup	bool	Determines if the element is a topic group.
NextElementDFS	Element	Denotes the pointer to the next Element component in the heirarchy.
PrevElementDFS	Element	Denotes the pointer to the prevoius Element component in the heirarchy.



Property name	Data Type	Description
ElementType	int	<p>The type of element. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_FO_CONTAINER (1)</li> <li>Constants.FV_FO_SYS_VAR (2)</li> <li>Constants.FV_FO_XREF (3)</li> <li>Constants.FV_FO_MARKER (4)</li> <li>Constants.FV_FO_FOOTNOTE (5)</li> <li>Constants.FV_FO_GRAPHIC (6)</li> <li>Constants.FV_FO_EQN (7)</li> <li>Constants.FV_FO_TBL (8)</li> <li>Constants.FV_FO_TBL_TITLE (9)</li> <li>Constants.FV_FO_TBL_HEADING (10)</li> <li>Constants.FV_FO_TBL_BODY (11)</li> <li>Constants.FV_FO_TBL_FOOTING (12)</li> <li>Constants.FV_FO_TBL_ROW (13)</li> <li>Constants.FV_FO_TBL_CELL (14)</li> <li>Constants.FV_FO_RUBIGROUP (15)</li> <li>Constants.FV_FO_RUBI (16)</li> </ul>
MatchingFirstPgfClauses	Objects	IDs of the first paragraph clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
FormatOverride	int	Returns 1 if the element has a format override. Else, returns 0.
ElementMarkedForNamedDestination	int	Used for generatig PDF. If 1, this element will have a corresponding Named Destination in the generated PDF.
MatchingLastPgfClauses	Objects	IDs of the last paragraph clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.

Property name	Data Type	Description
MatchingObjectClauses	Objects	IDs of the object clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
MatchingPrefixClauses	Objects	IDs of the prefix clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
MatchingSuffixClauses	Objects	IDs of the suffix clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
MatchingTextClauses	Objects	IDs of the text clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
TextRange	<i>TextRange</i>	Denotes the Text range that the element encompasses (see the explanation below).
TopicRefAbsolutePath	string	Denotes the absolute file path for the topic reference
Unique	int	Unique ID of the element.
UserString	string	A string to which clients can store private data.
BookComponent	<i>BookComponent</i>	Component file in book (FO_BookComponent ID).
ElementDef	<i>ElementDef</i>	Element's element definition (FO_ElementDef ID).
FirstChildElement	<i>Element</i>	Denote the element's first child element (FO_Element ID) if the element is a container, .
LastChildElement	<i>Element</i>	Denotes the element's last child element (FO_Element ID) if the element is a container.
NextSiblingElement	<i>Element</i>	Element's next sibling element (FO_Element ID).

Property name	Data Type	Description
NextDITAConrefElementInDoc	string	The next Conref element in the document
NextDITAXrefElementInDoc	string	The next XRef element in the document
NextDITALinkElementInDoc	string	The next Link element in the document
NextDITATopicrefElementInDoc	string	The next Topicref element in the document
NextDITATopicsetrefElementInDoc	string	The next Topicsetref element in the document

Property name	Data Type	Description
Object	FMObject	<p>ID of the object that an element contains. The type of object the ID specifies depends on the element definition as follows:</p> <ul style="list-style-type: none"> <li>• Constants.FV_FO_SYS_VAR (2): FO_Var</li> <li>Constants.FV_FO_XREF (3): FO_X-Ref</li> <li>Constants.FV_FO_MARKER (4): FO_Marker</li> <li>Constants.FV_FO_FOOTNOTE (5): FO_Fn</li> <li>Constants.FV_FO_GRAPHIC (6): FO_AFrame (containing the graphic)</li> <li>Constants.FV_FO_EQN (7): FO_A-Frame (containing the equation)</li> <li>Constants.FV_FO_TBL (8): FO_Tbl</li> <li>Constants.FV_FO_TBL_TITLE (9): FO_Tbl</li> <li>Constants.FV_FO_TBL_HEADING (10): FO_Tbl</li> <li>Constants.FV_FO_TBL_BODY (11): FO_Tbl</li> <li>Constants.FV_FO_TBL_FOOTING (12): FO_Tbl</li> <li>Constants.FV_FO_TBL_ROW (13): FO_Row</li> <li>Constants.FV_FO_TBL_CELL (14): FO_Cell</li> <li>Constants.FV_FO_RUBIGROUP (15): FO_Rubi</li> <li>Constants.FV_FO_RUBI (16): FO_Rubi</li> </ul>
ParentElement	<i>Element</i>	Element's parent element (FO_Element ID).
PrevSiblingElement	<i>Element</i>	Element's previous sibling element (FO_Element ID).

Property name	Data Type	Description
Namespace	<i>Strings</i>	Prefix/path pairs defining namespaces for the element. This list must contain an even number of strings. For example: <code>prefix1, path1prefix2, path2</code>
NamespaceScope	<i>Element</i>	The ID of element which declares the namespace that is used to define the current element.
NumNamespaces	<code>int</code>	The number of namespaces declared in the current element. This property is Readonly.
AllowAsSpecialCase	<code>int</code>	1 if the element is treated as a special case.
AttributeValueInvalid	<code>int</code>	1 if the element contains an attribute value that is invalid.
BookComponentMissing	<code>int</code>	1 if a component file is missing from a book.
ElementsUndefined	<code>int</code>	1 if the element is undefined.
ErrorInBookComponent	<code>int</code>	1 if there is a validation error for a component in a book.
ContentsLooselyValid	<code>int</code>	1 if the content is loosely valid (it has some missing elements).
ContentsStrictlyValid	<code>int</code>	1 if the content of the element is strictly valid.
ContentMustBeEmpty	<code>int</code>	1 if the element cannot have any content.
ContentNeededAtBegin	<code>int</code>	1 if content is needed at the beginning of the element.
ContentNeededAtEnd	<code>int</code>	1 if content is needed at end of the element. <code>ContentNeededAtEnd</code> is obsolete, but is supported for backward compatibility.
ElementsExcludedInContext	<code>int</code>	1 if the element is excluded.
ElementsInvalidInParent	<code>int</code>	1 if the element cannot occur anywhere in its current parent.

Property name	Data Type	Description
ElementIsInvalidInPosition	int	1 if the element is invalid in its current position.
HoleBeforeElement	int	1 if there are one or more missing elements before the element within the same parent.
InvalidHighestLevel	int	1 if the element cannot be the highest-level element in the flow.
NextInvalidElement	<i>Element</i>	Next invalid element in the document (FO_Element ID).
TextIsInvalidInElement	int	1 if the element contains only text and the element definition disallows it. <i>TextIsInvalidInElement</i> is obsolete and is no longer supported.

Property name	Data Type	Description
ValidationFlags	int	<p>Bit flags specifying the element's validity.</p> <p>To determine all the ways in which an element is invalid without querying all the validation properties, query the ValidationFalgs property.</p> <p>Each bit flag in the returned value represents the value of the validation property with the corresponding name. For example, if the <code>ElementTypeMismatch</code> property is <code>True</code>, the <code>Constants.FV_ELEM_TYPE_MISMATCH (0x0002)</code> flag is set.</p> <p>The various bit flags are as follows:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_ELEM_UNDEFINED (0x0001)</code> <i>Constants.FV_ELEM_TYPE_MISMATCH (0x0002)</i> <i>Constants.FV_ELEM_EXCLUDED (0x0004)</i> <i>Constants.FV_ELEM_INVALID_IN_PARENT (0x0008)</i> <i>Constants.FV_ELEM_INVALID_AT_POSITION (0x0010)</i> <i>Constants.FV_ELEM_HAS_TEXT_INVALID (0x0020)</i> <i>Constants.FV_ELEM_CONTENT_MUST_BE_EMPTY (0x0040)</i> <i>Constants.FV_ELEM_MISSING_CONTENT_BEFORE (0x0080)</i> <i>Constants.FV_ELEM_MISSING_CONTENT_AT_BEG (0x0100)</i> <i>Constants.FV_ELEM_MISSING_CONTENT_AT_END (0x0200)</i> <i>Constants.FV_ELEM_NOT_VALID_AS_ROOT (0x0400)</i> <i>Constants.FV_ELEM_BOOK_COMP_MISSING (0x0800)</i> <i>Constants.FV_ELEM_BOOK_COMP_INVALID (0x1000)</i></li> </ul>

Property name	Data Type	Description
Direction	int	<ul style="list-style-type: none"> <li>Constants.FV_ELEM_ATTRVAL_REQUIRED (0x2000)</li> <li>Constants.FV_ELEM_ATTRVAL_INVALID (0x4000)</li> <li>Constants.FV_ELEM_CONTENT_STRICTLY_VALID (0x10000)</li> <li>Constants.FV_ELEM_CONTENT_LOOSELY_VALID (0x20000)</li> </ul> Set or get the direction of the document. Possible values: <ul style="list-style-type: none"> <li>Constants.FV_DIR_Inherit - Inherit the direction of the parent</li> <li>Constants.FV_DIR_LTR - Left-to-right</li> <li>FV_DIR_RTL - Right-to-left</li> </ul>
ResolvedDirection	int	Get the inherited direction of the document. Possible values: <ul style="list-style-type: none"> <li>Constants.FV_DIR_LTR - Left-to-right</li> <li>FV_DIR_RTL - Right-to-left</li> </ul>

## Element methods

[Delete](#), [GetAttributes](#), [GetProps](#), [GetText](#), [ObjectValid](#), [SetProps](#).

## ElementCatalogEntries

An Array of [ElementCatalogEntry](#) objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## ElementCatalogEntries methods

[concat](#), [pop](#), [push](#).



## ElementCatalogEntry

Property name	Data Type	Description
obj	<i>ElementDef</i>	Denotes the ID of the element definition.
flags	uint	<p>The type of validation returned. The value is one of the following:</p> <ul style="list-style-type: none"> <li>Constants.FV_STRICTLY_VALID (0x01) - Catalog entry is strictly valid.</li> </ul> <p><i>Constants.FV_LOOSELY_VALID (0x02) - Catalog entry is loosely valid.</i></p> <p><i>Constants.FV_ALTERNATIVE (0x04) - Catalog entry is an alternative.</i></p> <p><i>Constants.FV_INCLUSION (0x08) - Catalog entry is valid because it is an inclusion.</i></p> <p><b>NOTE:</b> If none of the flags are set, then the element is invalid at the current position.</p>

### ElementCatalogEntry methods

*ElementCatalogEntry.*

## ElementDef

Property name	Data Type	Description
AttributeDefs	<i>AttributeDefs</i>	The element definition's attribute definitions.
BannerText	string	Property of element definition object, FO_ElementDef. Users can set, change or query the banner text associated with an element definition object.

Property name	Data Type	Description
Comment	string	Text string of the comment.
InitStructurePattern	string	The initial structure pattern; for table elements, a comma delimited string that specifies the necessary child elements to automatically insert.
ElementDefType	int	Type of formatter object represented by the element with element definition. Constants.FV_FO_CONTAINER (1) identifies a container element. Other values identify object (non-container) elements. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_FO_UNSPECIFIED (0)</li> <li>Constants.FV_FO_CONTAINER (1)</li> <li>Constants.FV_FO_SYS_VAR (2)</li> <li>Constants.FV_FO_XREF (3)</li> <li>Constants.FV_FO_MARKER (4)</li> <li>Constants.FV_FO_FOOTNOTE (5)</li> <li>Constants.FV_FO_GRAPHIC (6)</li> <li>Constants.FV_FO_EQN (7)</li> <li>Constants.FV_FO_TBL (8)</li> <li>Constants.FV_FO_TBL_TITLE (9)</li> <li>Constants.FV_FO_TBL_HEADING (10)</li> <li>Constants.FV_FO_TBL_BODY (11)</li> <li>Constants.FV_FO_TBL_FOOTING (12)</li> <li>Constants.FV_FO_TBL_ROW (13)</li> <li>Constants.FV_FO_TBL_CELL (14)</li> </ul>
ElementInCatalog	int	1 if the element is in the Element Catalog.
ElementPgfFormat	string	The name of the paragraph format applied to the element.
Exclusions	<i>Strings</i>	List of excluded elements.

Property name	Data Type	Description
FirstPgfRules	Objects	The IDs of the first paragraph format rules (FO_FmtRule IDs). <i>NOTE: To set the format rules for an element definition, use <code>NewFmtRuleObject()</code>.</i>
GeneralRule	string	Text of the element's general rule.
GeneralRuleErrorOffsets	<i>Ints</i>	Contains the error offsets (two positions are specified only if the content rule is ambiguous).
Inclusions	<i>Strings</i>	List of included elements.
LastPgfRules	Objects	The IDs of the last paragraph format rules (FO_FmtRule IDs).
Name	string	Name of the element definition.
NextElementDefInDoc	<i>ElementDef</i>	Next element definition in the document's list of element definitions (FO_ElementDef ID).
ObjectFmtRules	Objects	The IDs of the object format rules (FO_FmtRule IDs).
PrefixRules	Objects	The IDs of the prefix format rules (FO_FmtRule IDs).
SuffixRules	Objects	The IDs of the suffix format rules (FO_FmtRule IDs).
TextFmtRules	Objects	The IDs of the text format rules (FO_FmtRule IDs).
ValidHighestLevel	int	1 if the element can be used as the highest-level element for a flow.
ElementDataType	int	Retrieves/sets the data type of Element. The possible values are one of: <ul style="list-style-type: none"> <li>Constants.FV_DT_UNSPECIFIED(0)</li> <li>Constants.FV_DT_INTEGER(1)</li> <li>Constants.FV_DT_FLOAT(2)</li> </ul>
AllowedDataRange		Defines the permissible range of values for the values defined in the <code>ElementDataType</code> property.

Property name	Data Type	Description
DescriptiveTag	string	A small description about the element. If the user specifies this in the EDD, then the element description can be viewed in the Element Catalog of the structured document.
ElementDescription	string	Reserved for future use.
AlsoInsert	<i>Strings</i>	The list of the tags of child elements that are automatically inserted when an element is initially added.
AlsoInserts	<i>TypedVal</i>	The list of list of the tags of child elements that are automatically inserted when an element is initially added. For example: Setting the <code>AlsoInserts</code> property of element definition of A to <code>[[a1, a11],[a2, a21]]</code> , will automatically insert all child elements of A [a1, a2, a11 and a21] when A is inserted.

## ElementDef methods

*Delete, ElementDefIsText, GetAttributeDefs, GetProps, NewElement, NewElementInHierarchy, NewFirstPgfRules, NewLastPgfRules, NewObjectFmtRules, NewPrefixRules, NewSuffixRules, NewTextFmtRules, ObjectValid, SetProps, WrapElement.*

## ElementLoc

Property name	Data Type	Description
parent	<i>Element</i>	ID of the parent element.
child	<i>Element</i>	ID of the child element.
offset	int	Offset within the parent or child element.

## ElementLoc methods

[ElementLoc](#).

## ElementRange

Property name	Data Type	Description
beg	<a href="#">ElementLoc</a>	Beginning of the element range.
end	<a href="#">ElementLoc</a>	End of the element range.

## ElementRange methods

[ElementRange](#).

## Ellipse

Property name	Data Type	Description
RectangleIsSmoothed	int	1 if smoothing is enabled. This property is always 1 for Ellipse objects.
Angle	int	Angle of rotation of the ellipse.
ArrowBaseAngle	int	Arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to False.
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Arrowhead tip angle in degrees.

Property name	Data Type	Description
ArrowType	int	Arrowhead style. The value is one of the following: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Denotes the border width. The permissible range is from 0.015 points to 360 points.
Color	<i>Color</i>	The spot color.
Dash	<i>Metrics</i>	The dash style. The Dash property specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	The fill pattern (numbers between 0 and 15). The ESTK provides constants for the following fill patterns: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object (AFrame or UnAnchoredFrame).
GraphicCantBeSelected	int	1 if the graphic object cannot be selected; 0 otherwise.
GraphicsSelected	int	1 if the graphic object is selected; 0, otherwise.

Property name	Data Type	Description
GroupParent	<i>Group</i>	Group that the object is in (FO_Group ID). Anchored and unanchored frames do not have this property.
HeadArrow	int	1 if the line has an arrow head; otherwise, 0.
Height	int	Denotes the height of the object. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Type of line end. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Distance from the left side of the parent frame (–216 inches to 216 inches). <b>NOTE:</b> For anchored frames, the distance is automatically calculated from the left side of the page frame. You cannot manually set this property for anchored frames.
LocY	int	Distance from the top of the parent frame (–216 inches to 216 inches). <b>NOTE:</b> For anchored frames, the distance is automatically calculated from the top of the page frame. You cannot manually set this property for anchored frames.
NextGraphicInDoc	FMOBJECT	Next graphic object in the document.

Property name	Data Type	Description
NextGraphicInFrame	FObject	Next graphic object in the frame.
NextGraphicInGroup	FObject	Next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Next selected graphic object in document.
ObjectAttributes	<i>Strings</i>	A list of strings, each string expressing an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <code>&lt;tag&gt;string_text&lt;/tag&gt;</code> . You can use the backslash to escape characters such as <code>\n</code> , <code>\r</code> , or <code>\t</code> for newline, carriage return, and tab, respectively. For a backslash character in the string text, type <code>\\</code> .
Overprint	int	Specifies the overprint settings for the objec. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	The fill pattern (numbers between 0 and 15). The ESTK provides constants for the following fill patterns: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Previous graphic object in the frame.
PrevGraphicInGroup	FObject	Previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>



Property name	Data Type	Description
RunaroundGap	int	Denotes the width of the runaround gap if the object is a runaround object.
TailArrow	int	1 if arrowhead is enabled at the beginning of the line. 0, otherwise.
TintPercent	int	Denotes the tint percentage.
Unique	int	The graphic object's UID.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## Ellipse methods

*Delete, GetProps, ObjectValid, SetProps.*

## FCodes

Property name	Data Type
FCodes.NULLINPUT	NULLINPUT <b>Value-1</b>
FCodes.KBD_INPUT	KBD_INPUT <b>Value1</b>
FCodes.START_DIALOG	START_DIALOG <b>Value0xC100</b>
FCodes.END_DIALOG	END_DIALOG <b>Value0xC200</b>
FCodes.START_WINDOW	START_WINDOW <b>Value0xC400</b>
FCodes.END_WINDOW	END_WINDOW <b>Value0xC500</b>
FCodes.LEAVE_DOC	LEAVE_DOC <b>Value0xC600</b>
FCodes.SERVER_CANCEL	SERVER_CANCEL <b>Value0xC001</b>

Property name	Data Type
FCodes.SERVER_QUERY	SERVER_QUERY <b>Value0xC002</b>
FCodes.MENU_BARSTART	MENU_BARSTART <b>Value0xC301</b>
FCodes.MENU_WAITBIT	MENU_WAITBIT <b>Value0x0010</b>
FCodes.MENU_DOWNWAIT	MENU_DOWNWAIT <b>Value0xC312</b>
FCodes.MENU_RIGHTWAIT	MENU_RIGHTWAIT <b>Value0xC313</b>
FCodes.MENU_DOWN	MENU_DOWN <b>Value0xC304</b>
FCodes.MENU_RIGHT	MENU_RIGHT <b>Value0xC305</b>
FCodes.MENU_DONE	MENU_DONE <b>Value0xC30F</b>
FCodes.KBD_GBL_START	KBD_GBL_START <b>Value0x100</b>
FCodes.CSR_HOME	CSR_HOME <b>Value0x100</b>
FCodes.CSR_UP	CSR_UP <b>Value0x101</b>
FCodes.CSR_DOWN	CSR_DOWN <b>Value0x102</b>
FCodes.CSR_RIGHT	CSR_RIGHT <b>Value0x103</b>
FCodes.CSR_LEFT	CSR_LEFT <b>Value0x104</b>
FCodes.CSR_BOL	CSR_BOL <b>Value0x105</b>
FCodes.CSR_EOL	CSR_EOL <b>Value0x106</b>
FCodes.CSR_BOW	CSR_BOW <b>Value0x107</b>

Property name	Data Type
FCodes.CSR_EOW	CSR_EOW <b>Value0x108</b>
FCodes.CSR_BOS	CSR_BOS <b>Value0x109</b>
FCodes.CSR_EOS	CSR_EOS <b>Value0x10A</b>
FCodes.CSR_BOP	CSR_BOP <b>Value0x10B</b>
FCodes.CSR_EOP	CSR_EOP <b>Value0x10C</b>
FCodes.CSR_TOTR	CSR_TOTR <b>Value0x10D</b>
FCodes.CSR_BOTR	CSR_BOTR <b>Value0x10E</b>
FCodes.CSR_TOP	CSR_TOP <b>Value0x10F</b>
FCodes.CSR_BOT	CSR_BOT <b>Value0x110</b>
FCodes.DEL_CHARBWD	DEL_CHARBWD <b>Value0x112</b>
FCodes.DEL_CHARFWD	DEL_CHARFWD <b>Value0x113</b>
FCodes.DEL_BOW	DEL_BOW <b>Value0x114</b>
FCodes.DEL_EOW	DEL_EOW <b>Value0x115</b>
FCodes.DEL_EOL	DEL_EOL <b>Value0x116</b>
FCodes.DEL_EOS	DEL_EOS <b>Value0x117</b>
FCodes.DEL_SEL	DEL_SEL <b>Value0x118</b>
FCodes.DEL_BOL	DEL_BOL <b>Value0x119</b>

Property name	Data Type
FCodes.KBD_KERNUP	KBD_KERNUP <b>Value0x11A</b>
FCodes.KBD_KERNDOWN	KBD_KERNDOWN <b>Value0x11B</b>
FCodes.KBD_KERNLEFT	KBD_KERNLEFT <b>Value0x11C</b>
FCodes.KBD_KERNRIGHT	KBD_KERNRIGHT <b>Value0x11D</b>
FCodes.KBD_KERNHOME	KBD_KERNHOME <b>Value0x11E</b>
FCodes.KBD_KERNUP6	KBD_KERNUP6 <b>Value0x121</b>
FCodes.KBD_KERNDOWN6	KBD_KERNDOWN6 <b>Value0x122</b>
FCodes.KBD_KERNLEFT6	KBD_KERNLEFT6 <b>Value0x123</b>
FCodes.KBD_KERNRIGHT6	KBD_KERNRIGHT6 <b>Value0x124</b>
FCodes.KBD_OBJJUP	KBD_OBJJUP <b>Value0x125</b>
FCodes.KBD_OBJDOWN	KBD_OBJDOWN <b>Value0x126</b>
FCodes.KBD_OBJLEFT	KBD_OBJLEFT <b>Value0x127</b>
FCodes.KBD_OBJRIGHT	KBD_OBJRIGHT <b>Value0x128</b>
FCodes.CSR_NEXT_BOW	CSR_NEXT_BOW <b>Value0x140</b>
FCodes.CSR_NEXT_BOS	CSR_NEXT_BOS <b>Value0x141</b>
FCodes.CSR_NEXT_BOP	CSR_NEXT_BOP <b>Value0x142</b>
FCodes.CSR_FIRST_COL	CSR_FIRST_COL <b>Value0x143</b>

Property name	Data Type
FCodes.CSR_NEXT_COL	CSR_NEXT_COL <b>Value0x144</b>
FCodes.CSR_NEXT_ELEMENT	CSR_NEXT_ELEMENT <b>Value0x145</b>
FCodes.CSR_PREV_ELEMENT	CSR_PREV_ELEMENT <b>Value0x146</b>
FCodes.CSR_BOE	CSR_BOE <b>Value0x149</b>
FCodes.CSR_EOE	CSR_EOE <b>Value0x14A</b>
FCodes.CSR_BEFORE_ELEMENT	CSR_BEFORE_ELEMENT <b>Value0x14B</b>
FCodes.CSR_INT0_CHILD	CSR_INT0_CHILD <b>Value0x14D</b>
FCodes.CSR_PREV_EOS	CSR_PREV_EOS <b>Value0x937</b>
FCodes.CSR_PREV_EOW	CSR_PREV_EOW <b>Value0x938</b>
FCodes.DEL_WORD_START	DEL_WORD_START <b>Value0x160</b>
FCodes.DEL_WORD_END	DEL_WORD_END <b>Value0x161</b>
FCodes.DEL_NEXT_SS	DEL_NEXT_SS <b>Value0x162</b>
FCodes.DEL_BOS	DEL_BOS <b>Value0x163</b>
FCodes.KBD_NUMLOCK	KBD_NUMLOCK <b>Value0x170</b>
FCodes.KBD_GBL_END	KBD_GBL_END <b>Value0x1FF</b>
FCodes.KBD_BACKTAB	KBD_BACKTAB <b>Value0x220</b>
FCodes.KBD_SHFTSPACE	KBD_SHFTSPACE <b>Value0x221</b>

Property name	Data Type
FCodes.KBD_FIRSTTAB	KBD_FIRSTTAB <b>Value0x222</b>
FCodes.KBD_TABLE_TAB	KBD_TABLE_TAB <b>Value0x223</b>
FCodes.KBD_TAB	KBD_TAB <b>Value0x219</b>
FCodes.KBD_XCHARS	KBD_XCHARS <b>Value0x224</b>
FCodes.KBD_SOFTHYPHEN	KBD_SOFTHYPHEN <b>Value0x225</b>
FCodes.KBD_DONTHYPHEN	KBD_DONTHYPHEN <b>Value0x226</b>
FCodes.KBD_HARDHYPHEN	KBD_HARDHYPHEN <b>Value0x227</b>
FCodes.KBD_HARDSPACE	KBD_HARDSPACE <b>Value0x228</b>
FCodes.KBD_HARDRETURN	KBD_HARDRETURN <b>Value0x229</b>
FCodes.KBD_NUMSPACE	KBD_NUMSPACE <b>Value0x22A</b>
FCodes.KBD_THINSPACE	KBD_THINSPACE <b>Value0x22B</b>
FCodes.KBD_EMSPACE	KBD_EMSPACE <b>Value0x22C</b>
FCodes.KBD_ENSPACE	KBD_ENSPACE <b>Value0x22D</b>
FCodes.KBD_OPENLINE	KBD_OPENLINE <b>Value0x22E</b>
FCodes.KBD_RETURN	KBD_RETURN <b>Value0x22F</b>
FCodes.KBD_FPREV	KBD_FPREV <b>Value0x230</b>
FCodes.KBD_FNEXT	KBD_FNEXT <b>Value0x231</b>

Property name	Data Type
FCodes.KBD_ROMCE	KBD_ROMCE <b>Value0x232</b>
FCodes.KBD_RGLOBAL	KBD_RGLOBAL <b>Value0x233</b>
FCodes.KBD_RANDF	KBD_RANDF <b>Value0x234</b>
FCodes.KBD_SETSEARCH	KBD_SETSEARCH <b>Value0x235</b>
FCodes.HIGH_CHAR	HIGH_CHAR <b>Value0x240</b>
FCodes.HIGH_WORD	HIGH_WORD <b>Value0x241</b>
FCodes.HIGH_LINE	HIGH_LINE <b>Value0x242</b>
FCodes.HIGH_SENT	HIGH_SENT <b>Value0x243</b>
FCodes.HIGH_PGF	HIGH_PGF <b>Value0x244</b>
FCodes.HIGH_SHL	HIGH_SHL <b>Value0x245</b>
FCodes.HIGH_SHR	HIGH_SHR <b>Value0x246</b>
FCodes.HIGH_CLEAR	HIGH_CLEAR <b>Value0x247</b>
FCodes.HIGH_FLOW	HIGH_FLOW <b>Value0x248</b>
FCodes.HIGH_ELEMENT	HIGH_ELEMENT <b>Value0x249</b>
FCodes.KBD_ABORT	KBD_ABORT <b>Value0x250</b>
FCodes.KBD_CAPTURE	KBD_CAPTURE <b>Value0x251</b>
FCodes.KBD_ECAPTURE	KBD_ECAPTURE <b>Value0x256</b>

Property name	Data Type
FCodes.KBD_RECORD	KBD_RECORD <b>Value0x252</b>
FCodes.KBD_GETTRIGGER	KBD_GETTRIGGER <b>Value0x253</b>
FCodes.KBD_ABORT_DLGS	KBD_ABORT_DLGS <b>Value0x254</b>
FCodes.KBD_WAIT_DLGS	KBD_WAIT_DLGS <b>Value0x255</b>
FCodes.TXT_BOLD	TXT_BOLD <b>Value0x260</b>
FCodes.TXT_ITALIC	TXT_ITALIC <b>Value0x261</b>
FCodes.TXT_UNDERLINE	TXT_UNDERLINE <b>Value0x262</b>
FCodes.TXT_PLAIN	TXT_PLAIN <b>Value0x263</b>
FCodes.TXT_SUPER	TXT_SUPER <b>Value0x264</b>
FCodes.TXT_SUB	TXT_SUB <b>Value0x265</b>
FCodes.TXT_NORMAL	TXT_NORMAL <b>Value0x266</b>
FCodes.TXT_INCSIZE	TXT_INCSIZE <b>Value0x267</b>
FCodes.TXT_DECSIZE	TXT_DECSIZE <b>Value0x268</b>
FCodes.TXT_SELINCSIZE	TXT_SELINCSIZE <b>Value0x11F</b>
FCodes.TXT_SELDECSIZE	TXT_SELDECSIZE <b>Value0x120</b>
FCodes.TXT_SQUEEZE	TXT_SQUEEZE <b>Value0x269</b>
FCodes.TXT_SPREAD	TXT_SPREAD <b>Value0x26A</b>



Property name	Data Type
FCodes.TXT_NOSTRETCH	TXT_NOSTRETCH <b>Value0x130</b>
FCodes.TXT_LESSSTRETCH	TXT_LESSSTRETCH <b>Value0x131</b>
FCodes.TXT_MORESTRETCH	TXT_MORESTRETCH <b>Value0x132</b>
FCodes.TXT_BAM	TXT_BAM <b>Value0x26B</b>
FCodes.TXT_STRIKEOUT	TXT_STRIKEOUT <b>Value0x26C</b>
FCodes.TXT_DEFAULT	TXT_DEFAULT <b>Value0x26D</b>
FCodes.TXT_OVERLINE	TXT_OVERLINE <b>Value0x26E</b>
FCodes.TXT_CHANGEBAR	TXT_CHANGEBAR <b>Value0x26F</b>
FCodes.TXT_KERN	TXT_KERN <b>Value0x270</b>
FCodes.TXT_OUTLINE	TXT_OUTLINE <b>Value0x271</b>
FCodes.TXT_SHADOW	TXT_SHADOW <b>Value0x272</b>
FCodes.TXT_MINICAPS	TXT_MINICAPS <b>Value0x273</b>
FCodes.TXT_INITCAPS	TXT_INITCAPS <b>Value0x274</b>
FCodes.TXT_UPPERCASE	TXT_UPPERCASE <b>Value0x275</b>
FCodes.TXT_LOWERCASE	TXT_LOWERCASE <b>Value0x276</b>
FCodes.TXT_NORMALCASE	TXT_NORMALCASE <b>Value0x277</b>
FCodes.TXT_DBLUNDERLINE	TXT_DBLUNDERLINE <b>Value0x278</b>

Property name	Data Type
FCodes.TXT_NUMUNDERLINE	TXT_NUMUNDERLINE <b>Value0x279</b>
FCodes.TXT_TSUME	TXT_TSUME <b>Value0x2A5</b>
FCodes.TXT_SELBOLD	TXT_SELBOLD <b>Value0x27A</b>
FCodes.TXT_SELITALIC	TXT_SELITALIC <b>Value0x27B</b>
FCodes.TXT_SELUNDERLINE	TXT_SELUNDERLINE <b>Value0x27C</b>
FCodes.TXT_SELPLAIN	TXT_SELPLAIN <b>Value0x27D</b>
FCodes.TXT_7	TXT_7 <b>Value0xC00</b>
FCodes.TXT_9	TXT_9 <b>Value0xC01</b>
FCodes.TXT_10	TXT_10 <b>Value0xC02</b>
FCodes.TXT_12	TXT_12 <b>Value0xC03</b>
FCodes.TXT_14	TXT_14 <b>Value0xC04</b>
FCodes.TXT_18	TXT_18 <b>Value0xC05</b>
FCodes.TXT_24	TXT_24 <b>Value0xC06</b>
FCodes.TXT_USIZE	TXT_USIZE <b>Value0xC07</b>
FCodes.TXT_OTHERSIZE	TXT_OTHERSIZE <b>Value0xC08</b>
FCodes.TXT_FAMILY_AND_VARIATION	TXT_FAMILY_AND_VARIATION <b>Value0xC09</b>
FCodes.TXT_FAMILY_AND_VARIATION_PLATFORM	TXT_FAMILY_AND_VARIATION_PLATFORM <b>Value0xC0A</b>

Property name	Data Type
FCodes.PGF_APPLY_TAG	PGF_APPLY_TAG <b>Value0xC10</b>
FCodes.CHAR_APPLY_TAG	CHAR_APPLY_TAG <b>Value0xC11</b>
FCodes.MENU_EXPOSE_WIN	MENU_EXPOSE_WIN <b>Value0xC12</b>
FCodes.MRU_SELECT_FILE	MRU_SELECT_FILE <b>Value0xC13</b>
FCodes.MENU_IMPORT_FILE	MENU_IMPORT_FILE <b>Value0xC14</b>
FCodes.KBD_EXPORT_GRAPHIC	KBD_EXPORT_GRAPHIC <b>Value0xC15</b>
FCodes.KBD_EXPORT_DOCUMENT	KBD_EXPORT_DOCUMENT <b>Value0xC16</b>
FCodes.TXT_FONT	TXT_FONT <b>Value0xC17</b>
FCodes.PGF_APPLY_CAT_TO_SEL	PGF_APPLY_CAT_TO_SEL <b>Value0xF3D</b>
FCodes.CHAR_APPLY_CAT_TO_SEL	CHAR_APPLY_CAT_TO_SEL <b>Value0xF3E</b>
FCodes.PGF_HYPHENON	PGF_HYPHENON <b>Value0x27E</b>
FCodes.PGF_HYPHENOFF	PGF_HYPHENOFF <b>Value0x27F</b>
FCodes.PGF_INCLINE	PGF_INCLINE <b>Value0x280</b>
FCodes.PGF_DECLINE	PGF_DECLINE <b>Value0x281</b>
FCodes.PGF_CENTER	PGF_CENTER <b>Value0x282</b>
FCodes.PGF_LEFT	PGF_LEFT <b>Value0x283</b>
FCodes.PGF_RIGHT	PGF_RIGHT <b>Value0x284</b>

Property name	Data Type
FCodes.PGF_FULL	PGF_FULL <b>Value0x285</b>
FCodes.PGF_BAM	PGF_BAM <b>Value0x286</b>
FCodes.PGF_LINEFIX	PGF_LINEFIX <b>Value0x287</b>
FCodes.PGF_LINEFLOAT	PGF_LINEFLOAT <b>Value0x288</b>
FCodes.PGF_UNIFY	PGF_UNIFY <b>Value0x289</b>
FCodes.PGF_PAGETOP	PGF_PAGETOP <b>Value0x28A</b>
FCodes.PGF_COLTOP	PGF_COLTOP <b>Value0x28B</b>
FCodes.PGF_LEFTTOP	PGF_LEFTTOP <b>Value0x28C</b>
FCodes.PGF_RIGHTTOP	PGF_RIGHTTOP <b>Value0x28D</b>
FCodes.PGF_ANYPLACE	PGF_ANYPLACE <b>Value0x28E</b>
FCodes.KBD_ALIGN_TOP	KBD_ALIGN_TOP <b>Value0x290</b>
FCodes.KBD_ALIGN_MIDDLE	KBD_ALIGN_MIDDLE <b>Value0x291</b>
FCodes.KBD_ALIGN_BOTTOM	KBD_ALIGN_BOTTOM <b>Value0x292</b>
FCodes.KBD_OBALIGN_TOP	KBD_OBALIGN_TOP <b>Value0x129</b>
FCodes.KBD_OBALIGN_MIDDLE	KBD_OBALIGN_MIDDLE <b>Value0x12A</b>
FCodes.KBD_OBALIGN_BOTTOM	KBD_OBALIGN_BOTTOM <b>Value0x12B</b>
FCodes.KBD_OBALIGN_CENTER	KBD_OBALIGN_CENTER <b>Value0x12C</b>

Property name	Data Type
FCodes.KBD_OBALIGN_LEFT	KBD_OBALIGN_LEFT <b>Value0x12D</b>
FCodes.KBD_OBALIGN_RIGHT	KBD_OBALIGN_RIGHT <b>Value0x12E</b>
FCodes.PGF_SINGLE_SPACE	PGF_SINGLE_SPACE <b>Value0x293</b>
FCodes.PGF_ONEANDAHALF_SPACE	PGF_ONEANDAHALF_SPACE <b>Value0x294</b>
FCodes.PGF_DOUBLE_SPACE	PGF_DOUBLE_SPACE <b>Value0x295</b>
FCodes.PGF_SPACE_BETWEEN	PGF_SPACE_BETWEEN <b>Value0x296</b>
FCodes.PGF_LINE_SPACE	PGF_LINE_SPACE <b>Value0x297</b>
FCodes.PGF_UPDATE_ALL	PGF_UPDATE_ALL <b>Value0x298</b>
FCodes.PGF_NEW_FORMAT	PGF_NEW_FORMAT <b>Value0x299</b>
FCodes.KBD_HELP_INDEX	KBD_HELP_INDEX <b>Value0x2A0</b>
FCodes.KBD_HELP_KEYS	KBD_HELP_KEYS <b>Value0x2A1</b>
FCodes.KBD_HELP_SAMPLES	KBD_HELP_SAMPLES <b>Value0x2A2</b>
FCodes.KBD_HELP_OVERVIEW	KBD_HELP_OVERVIEW <b>Value0x2A3</b>
FCodes.KBD_HELP_ONLINEMANUALS	KBD_HELP_ONLINEMANUALS <b>Value0x2A4</b>
FCodes.KBD_HELP_WEBWORKS	KBD_HELP_WEBWORKS <b>Value0x2A6</b>
FCodes.KBD_CUSTOMNEW	KBD_CUSTOMNEW <b>Value0x2FF</b>
FCodes.KBD_NEW	KBD_NEW <b>Value0x300</b>

Property name	Data Type
FCodes.KBD_NEWBOOK	KBD_NEWBOOK <b>Value0x308</b>
FCodes.KBD_OPEN	KBD_OPEN <b>Value0x301</b>
FCodes.KBD_HEROIC_OPEN	KBD_HEROIC_OPEN <b>Value0x306</b>
FCodes.KBD_SILENT_OPEN	KBD_SILENT_OPEN <b>Value0x307</b>
FCodes.KBD_HELP	KBD_HELP <b>Value0x302</b>
FCodes.KBD_INFO	KBD_INFO <b>Value0x303</b>
FCodes.KBD_CSHELPMODE	KBD_CSHELPMODE <b>Value0x304</b>
FCodes.KBD_ABOUTPRODUCT	KBD_ABOUTPRODUCT <b>Value0x305</b>
FCodes.KBD_BOOKADDFILE	KBD_BOOKADDFILE <b>Value0x30A</b>
FCodes.KBD_BOOKEDITDEFINE	KBD_BOOKEDITDEFINE <b>Value0x30B</b>
FCodes.KBD_BOOKRENAMEFILE	KBD_BOOKRENAMEFILE <b>Value0x30C</b>
FCodes.KBD_BOOKDISPLAYFILENAME	KBD_BOOKDISPLAYFILENAME <b>Value0x30D</b>
FCodes.KBD_BOOKDISPLAYTEXT	KBD_BOOKDISPLAYTEXT <b>Value0x30E</b>
FCodes.KBD_BOOKCOMP_EXCLUDE	KBD_BOOKCOMP_EXCLUDE <b>Value0x4E0</b>
FCodes.KBD_BOOKADDFOLDER	KBD_BOOKADDFOLDER <b>Value0x601</b>
FCodes.KBD_BOOKCOMP_FILENAME	KBD_BOOKCOMP_FILENAME <b>Value0x607</b>
FCodes.KBD_BOOKCOMP_TEXT	KBD_BOOKCOMP_TEXT <b>Value0x608</b>

Property name	Data Type
FCodes.KBD_BOOKADDGROUP	KBD_BOOKADDGROUP <b>Value0x609</b>
FCodes.KBD_MOVE_UP	KBD_MOVE_UP <b>Value0x70A</b>
FCodes.KBD_MOVE_DOWN	KBD_MOVE_DOWN <b>Value0x70B</b>
FCodes.KBD_MOVE_LEFT	KBD_MOVE_LEFT <b>Value0x70C</b>
FCodes.KBD_MOVE_RIGHT	KBD_MOVE_RIGHT <b>Value0x70D</b>
FCodes.KBD_OPEN_COMPONENTS	KBD_OPEN_COMPONENTS <b>Value0x3EA</b>
FCodes.KBD_CLOSE_COMPONENTS	KBD_CLOSE_COMPONENTS <b>Value0x3EB</b>
FCodes.KBD_PRINT_COMPONENTS	KBD_PRINT_COMPONENTS <b>Value0x3EC</b>
FCodes.KBD_SAVE_COMPONENTS	KBD_SAVE_COMPONENTS <b>Value0x3ED</b>
FCodes.KBD_COMPONENT_PROPERTIES	KBD_COMPONENT_PROPERTIES <b>Value0x3EE</b>
FCodes.KBD_DELETE_FILE	KBD_DELETE_FILE <b>Value0x30F</b>
FCodes.KBD_COLLAPSE_ALL	KBD_COLLAPSE_ALL <b>Value0x3FA</b>
FCodes.KBD_EXPAND_ALL	KBD_EXPAND_ALL <b>Value0x3FB</b>
FCodes.SWITCH_TO_RESOURCEMANAGER	SWITCH_TO_RESOURCEMANAGER <b>Value0x9EA</b>
FCodes.SWITCH_TO_DOCUMENTVIEW	SWITCH_TO_DOCUMENTVIEW <b>Value0x9EB</b>
FCodes.KBD_SAVE	KBD_SAVE <b>Value0x310</b>
FCodes.KBD_SAVEAS	KBD_SAVEAS <b>Value0x311</b>

Property name	Data Type
FCodes.KBD_SAVEASPDFREVIEW	KBD_SAVEASPDFREVIEW <b>Value0x95A</b>
FCodes.KBD_SAVEASPDFREVIEW2	KBD_SAVEASPDFREVIEW2 <b>Value0x95B</b>
FCodes.KBD_SAVEASPDFUBIQ	KBD_SAVEASPDFUBIQ <b>Value0x95E</b>
FCodes.KBD_SAVEASPDFSHARE	KBD_SAVEASPDFSHARE <b>Value0x95F</b>
FCodes.KBD_SAVEASPDF	KBD_SAVEASPDF <b>Value0x950</b>
FCodes.KBD_SAVEASXML	KBD_SAVEASXML <b>Value0x951</b>
FCodes.KBD_MANCOND	KBD_MANCOND <b>Value0x952</b>
FCodes.KBD_CONDINDICATOR	KBD_CONDINDICATOR <b>Value0x953</b>
FCodes.KBD_ATTRCOND	KBD_ATTRCOND <b>Value0x954</b>
FCodes.KBD_PREVIEWFBA	KBD_PREVIEWFBA <b>Value0xB16</b>
FCodes.AMT_REGISRATION	AMT_REGISRATION <b>Value0x965</b>
FCodes.AMT_DEACTIVATION	AMT_DEACTIVATION <b>Value0x967</b>
FCodes.AMT_UPDATES	AMT_UPDATES <b>Value0x968</b>
FCodes.KBD_REVERT	KBD_REVERT <b>Value0x312</b>
FCodes.KBD_DOCINFO	KBD_DOCINFO <b>Value0x3DA</b>
FCodes.KBD_SETPOSTER	KBD_SETPOSTER <b>Value0x3DB</b>
FCodes.KBD_REMOVEPOSTER	KBD_REMOVEPOSTER <b>Value0x3DE</b>



Property name	Data Type
FCodes.KBD_PRINT	KBD_PRINT <b>Value0x313</b>
FCodes.KBD_IMPORT	KBD_IMPORT <b>Value0x314</b>
FCodes.KBD_GENERATE	KBD_GENERATE <b>Value0x3E1</b>
FCodes.KBD_USEFMTFROM	KBD_USEFMTFROM <b>Value0x316</b>
FCodes.KBD_KBMACRO	KBD_KBMACRO <b>Value0x317</b>
FCodes.KBD_SESSION	KBD_SESSION <b>Value0x318</b>
FCodes.KBD_PAGESETUP	KBD_PAGESETUP <b>Value0x319</b>
FCodes.KBD_OPENALL	KBD_OPENALL <b>Value0x31A</b>
FCodes.KBD_QUITALL	KBD_QUITALL <b>Value0x31B</b>
FCodes.KBD_SAVEALL	KBD_SAVEALL <b>Value0x31C</b>
FCodes.KBD_REPEATNEW	KBD_REPEATNEW <b>Value0x31D</b>
FCodes.KBD_PODLOCATION	KBD_PODLOCATION <b>Value0x31E</b>
FCodes.KBD_COMPARE	KBD_COMPARE <b>Value0x31F</b>
FCodes.KBD_UNDO	KBD_UNDO <b>Value0x320</b>
FCodes.KBD_REDO	KBD_REDO <b>Value0x935</b>
FCodes.KBD_HIST	KBD_HIST <b>Value0x936</b>
FCodes.KBD_REPEAT	KBD_REPEAT <b>Value0x939</b>

Property name	Data Type
FCodes.KBD_CUT	KBD_CUT <b>Value0x321</b>
FCodes.KBD_COPY	KBD_COPY <b>Value0x322</b>
FCodes.KBD_PASTE	KBD_PASTE <b>Value0x323</b>
FCodes.KBD_CLEAR	KBD_CLEAR <b>Value0x324</b>
FCodes.KBD_COPYFONT	KBD_COPYFONT <b>Value0x325</b>
FCodes.KBD_COPYPGF	KBD_COPYPGF <b>Value0x326</b>
FCodes.KBD_SELECTALL	KBD_SELECTALL <b>Value0x327</b>
FCodes.KBD_STUFF	KBD_STUFF <b>Value0x328</b>
FCodes.KBD_SEARCH	KBD_SEARCH <b>Value0x329</b>
FCodes.KBD_SPELLING	KBD_SPELLING <b>Value0x32A</b>
FCodes.KBD_CAPITAL	KBD_CAPITAL <b>Value0x32B</b>
FCodes.KBD_YANK	KBD_YANK <b>Value0x32C</b>
FCodes.KBD_SELECT_GENERATED_FILES	KBD_SELECT_GENERATED_FILES <b>Value0x32D</b>
FCodes.KBD_SELECT_NON_GENERATED_FILES	KBD_SELECT_NON_GENERATED_FILES <b>Value0x32E</b>
FCodes.KBD_SELECT_FM_FILES	KBD_SELECT_FM_FILES <b>Value0x315</b>
FCodes.KBD_SELECT_PRINTABLE_FILES	KBD_SELECT_PRINTABLE_FILES <b>Value0x9E2</b>
FCodes.KBD_SELECT_EXCLUDED_FILE	KBD_SELECT_EXCLUDED_FILE <b>Value0x9D1</b>

Property name	Data Type
FCodes.KBD_SELECT_NONEXCLUDED_FILE	KBD_SELECT_NONEXCLUDED_FILE <b>Value0x9D2</b>
FCodes.KBD_SELECT_CHAPTER_COMPONENTS	KBD_SELECT_CHAPTER_COMPONENTS <b>Value0x9D3</b>
FCodes.KBD_SELECT_SECTION_COMPONENTS	KBD_SELECT_SECTION_COMPONENTS <b>Value0x9D4</b>
FCodes.KBD_SELECT_SUBSECTION_COMPONENTS	KBD_SELECT_SUBSECTION_COMPONENTS <b>Value0x9D5</b>
FCodes.KBD_ALLCAP	KBD_ALLCAP <b>Value0x33A</b>
FCodes.KBD_ALLLOWER	KBD_ALLLOWER <b>Value0x33B</b>
FCodes.KBD_INITCAP	KBD_INITCAP <b>Value0x33C</b>
FCodes.KBD_ALLCAPH	KBD_ALLCAPH <b>Value0x369</b>
FCodes.KBD_ALLLOWERH	KBD_ALLLOWERH <b>Value0x36A</b>
FCodes.KBD_INITCAPH	KBD_INITCAPH <b>Value0x36B</b>
FCodes.KBD_THESAURUS	KBD_THESAURUS <b>Value0x3D0</b>
FCodes.KBD_THESAURUS_REPLACE	KBD_THESAURUS_REPLACE <b>Value0x3D3</b>
FCodes.KBD_CREATE_PUBLISHER	KBD_CREATE_PUBLISHER <b>Value0x3D5</b>
FCodes.KBD_SUBSCRIBE_TO	KBD_SUBSCRIBE_TO <b>Value0x3D6</b>
FCodes.KBD_LINK_BOUNDARIES	KBD_LINK_BOUNDARIES <b>Value0x3D7</b>
FCodes.KBD_FONTDESIGN	KBD_FONTDESIGN <b>Value0x330</b>
FCodes.KBD_PGFDESIGN	KBD_PGFDESIGN <b>Value0x331</b>

Property name	Data Type
FCodes.KBD_RUBIPROPS	KBD_RUBIPROPS <b>Value0x3DD</b>
FCodes.KBD_COLLAYOUT	KBD_COLLAYOUT <b>Value0x348</b>
FCodes.KBD_LINELAYOUT	KBD_LINELAYOUT <b>Value0x332</b>
FCodes.KBD_PAGESIZE	KBD_PAGESIZE <b>Value0x349</b>
FCodes.KBD_PAGINATION	KBD_PAGINATION <b>Value0xA40</b>
FCodes.KBD_PAGEBACK	KBD_PAGEBACK <b>Value0x34A</b>
FCodes.KBD_NEWMASTER	KBD_NEWMASTER <b>Value0xA0A</b>
FCodes.KBD_REORDERMASTER	KBD_REORDERMASTER <b>Value0xA0B</b>
FCodes.KBD_PAGEUPDATE	KBD_PAGEUPDATE <b>Value0xA00</b>
FCodes.KBD_CUST_TEXT_FRAME	KBD_CUST_TEXT_FRAME <b>Value0xA01</b>
FCodes.KBD_CONNECT_TEXT_FRAME	KBD_CONNECT_TEXT_FRAME <b>Value0x35B</b>
FCodes.KBD_CUTHEAD	KBD_CUTHEAD <b>Value0x35C</b>
FCodes.KBD_CURTAIL	KBD_CURTAIL <b>Value0x35D</b>
FCodes.KBD_CUTBOTH	KBD_CUTBOTH <b>Value0xA04</b>
FCodes.KBD_SPLIT	KBD_SPLIT <b>Value0x35A</b>
FCodes.KBD_SPLITR	KBD_SPLITR <b>Value0xA02</b>
FCodes.KBD_SPLITL	KBD_SPLITL <b>Value0xA03</b>

Property name	Data Type
FCodes.KBD_ROTPAGE_PLUS	KBD_ROTPAGE_PLUS <b>Value0x34E</b>
FCodes.KBD_ROTPAGE_MINUS	KBD_ROTPAGE_MINUS <b>Value0x34F</b>
FCodes.KBD_ROTPAGE_NORM	KBD_ROTPAGE_NORM <b>Value0xA05</b>
FCodes.KBD_NUMBERING	KBD_NUMBERING <b>Value0x333</b>
FCodes.KBD_CBARPRO	KBD_CBARPRO <b>Value0x334</b>
FCodes.KBD_FOOTNOTEPRO	KBD_FOOTNOTEPRO <b>Value0x335</b>
FCodes.KBD_TEXT_OPTIONS	KBD_TEXT_OPTIONS <b>Value0x337</b>
FCodes.KBD_COMBINED_FONTS	KBD_COMBINED_FONTS <b>Value0x338</b>
FCodes.KBD_ACROBAT_SETUP	KBD_ACROBAT_SETUP <b>Value0x36C</b>
FCodes.KBD_LINENUMPRO	KBD_LINENUMPRO <b>Value0x673</b>
FCodes.KBD_LINENUMTOGGLE	KBD_LINENUMTOGGLE <b>Value0x674</b>
FCodes.KBD_FIRSTPAGE	KBD_FIRSTPAGE <b>Value0x340</b>
FCodes.KBD_LASTPAGE	KBD_LASTPAGE <b>Value0x341</b>
FCodes.KBD_BODYPAGE	KBD_BODYPAGE <b>Value0x342</b>
FCodes.KBD_MASTERPAGE	KBD_MASTERPAGE <b>Value0x343</b>
FCodes.KBD_REFPAGE	KBD_REFPAGE <b>Value0x344</b>
FCodes.KBD_GOTOPAGE	KBD_GOTOPAGE <b>Value0x345</b>

Property name	Data Type
FCodes.KBD_ADDPAGE	KBD_ADDPAGE <b>Value0x346</b>
FCodes.KBD_DELETEPAGE	KBD_DELETEPAGE <b>Value0x347</b>
FCodes.KBD_FREEZE	KBD_FREEZE <b>Value0x34B</b>
FCodes.KBD_TEXTCOLPRO	KBD_TEXTCOLPRO <b>Value0x339</b>
FCodes.KBD_PREVPAGE	KBD_PREVPAGE <b>Value0x34C</b>
FCodes.KBD_NEXTPAGE	KBD_NEXTPAGE <b>Value0x34D</b>
FCodes.KBD_GOTOPAGEN	KBD_GOTOPAGEN <b>Value0x96E</b>
FCodes.KBD_GOTOIP	KBD_GOTOIP <b>Value0x96F</b>
FCodes.KBD_STRWIN_LEFTANCHOR	KBD_STRWIN_LEFTANCHOR <b>Value0x3DF</b>
FCodes.KBD_PAGEBREAK	KBD_PAGEBREAK <b>Value0x32F</b>
FCodes.KBD_ANCHOR	KBD_ANCHOR <b>Value0x350</b>
FCodes.KBD_FOOTNOTE	KBD_FOOTNOTE <b>Value0x351</b>
FCodes.KBD_REFERENCE	KBD_REFERENCE <b>Value0x352</b>
FCodes.KBD_VARIABLE	KBD_VARIABLE <b>Value0x353</b>
FCodes.KBD_INSET	KBD_INSET <b>Value0x354</b>
FCodes.KBD_HYPertext	KBD_HYPertext <b>Value0x359</b>
FCodes.KBD_MARKERS	KBD_MARKERS <b>Value0x355</b>

Property name	Data Type
FCodes.KBD_EQUATION	KBD_EQUATION <b>Value0x336</b>
FCodes.KBD_CONDTEXT	KBD_CONDTEXT <b>Value0x357</b>
FCodes.KBD_NEWMARKER	KBD_NEWMARKER <b>Value0x356</b>
FCodes.KBD_EDITMARKERTYPE	KBD_EDITMARKERTYPE <b>Value0x409</b>
FCodes.KBD_DELMARKERTYPE	KBD_DELMARKERTYPE <b>Value0x358</b>
FCodes.KBD_RENAMEMARKERTYPE	KBD_RENAMEMARKERTYPE <b>Value0x35F</b>
FCodes.KBD_ADDMARKERTYPE	KBD_ADDMARKERTYPE <b>Value0x41D</b>
FCodes.KBD_DELETEMARKER	KBD_DELETEMARKER <b>Value0x969</b>
FCodes.KBD_MARKERSPOD	KBD_MARKERSPOD <b>Value0x96A</b>
FCodes.KBD_XREFSPOD	KBD_XREFSPOD <b>Value0x96B</b>
FCodes.KBD_FONTPOD	KBD_FONTPOD <b>Value0x966</b>
FCodes.KBD_FONTREPLACEPOD	KBD_FONTREPLACEPOD <b>Value0x95C</b>
FCodes.KBD_HOTSPOTSPOD	KBD_HOTSPOTSPOD <b>Value0x4FF</b>
FCodes.KBD_NEWAFRAME	KBD_NEWAFRAME <b>Value0x96D</b>
FCodes.KBD_NEWHYPERTEXT	KBD_NEWHYPERTEXT <b>Value0x35E</b>
FCodes.KBD_VALIDATE_HYPERTEXT	KBD_VALIDATE_HYPERTEXT <b>Value0xF13</b>
FCodes.MENU_HYPERTEXT	MENU_HYPERTEXT <b>Value0xC18</b>

Property name	Data Type
FCodes.KBD_HYPRTXT_SHTCUT	KBD_HYPRTXT_SHTCUT <b>Value0x41E</b>
FCodes.WEB_GOTOADOBE	WEB_GOTOADOBE <b>Value0xC19</b>
FCodes.KBD_FMPIP	KBD_FMPIP <b>Value0x806</b>
FCodes.WEB_PREFERENCES	WEB_PREFERENCES <b>Value0xC1A</b>
FCodes.WEB_TOPISSUES	WEB_TOPISSUES <b>Value0xC1B</b>
FCodes.WEB_ADOBEHELP	WEB_ADOBEHELP <b>Value0xC22</b>
FCodes.WEB_CORPORATENEWS	WEB_CORPORATENEWS <b>Value0xC1D</b>
FCodes.WEB_REGISTRATION	WEB_REGISTRATION <b>Value0xC1E</b>
FCodes.WEB_FRAME_BOOKMARKS	WEB_FRAME_BOOKMARKS <b>Value0xC1F</b>
FCodes.WEB_ADOBE_BOOKMARKS	WEB_ADOBE_BOOKMARKS <b>Value0xC21</b>
FCodes.KBD_DUMPHYPertext	KBD_DUMPHYPertext <b>Value0xC20</b>
FCodes.KBD_VARCURPG	KBD_VARCURPG <b>Value0xA06</b>
FCodes.KBD_VARPGCOUNT	KBD_VARPGCOUNT <b>Value0xA07</b>
FCodes.KBD_VARCURDATE	KBD_VARCURDATE <b>Value0xA08</b>
FCodes.KBD_VAROTHER	KBD_VAROTHER <b>Value0xA09</b>
FCodes.KBD_EDITVARIABLE	KBD_EDITVARIABLE <b>Value0xA0C</b>
FCodes.KBD_FINDNEXT	KBD_FINDNEXT <b>Value0x955</b>



Property name	Data Type
FCodes.KBD_FINDPREV	KBD_FINDPREV <b>Value0x956</b>
FCodes.KBD_TRACKCHANGE	KBD_TRACKCHANGE <b>Value0x957</b>
FCodes.KBD_TRACKCHANGEDISABLE	KBD_TRACKCHANGEDISABLE <b>Value0x95D</b>
FCodes.KBD_ACCEPTCHANGE	KBD_ACCEPTCHANGE <b>Value0x958</b>
FCodes.KBD_REJECTCHANGE	KBD_REJECTCHANGE <b>Value0x959</b>
FCodes.KBD_ACCEPTCHANGEANDNEXT	KBD_ACCEPTCHANGEANDNEXT <b>Value0x99B</b>
FCodes.KBD_REJECTCHANGEANDNEXT	KBD_REJECTCHANGEANDNEXT <b>Value0x99C</b>
FCodes.KBD_ACCEPTALLCHANGE	KBD_ACCEPTALLCHANGE <b>Value0x960</b>
FCodes.KBD_REJECTALLCHANGE	KBD_REJECTALLCHANGE <b>Value0x961</b>
FCodes.KBD_PREVIEW_ACCEPTALL	KBD_PREVIEW_ACCEPTALL <b>Value0x962</b>
FCodes.KBD_PREVIEW_REJECTALL	KBD_PREVIEW_REJECTALL <b>Value0x963</b>
FCodes.KBD_PREVIEW_OFF	KBD_PREVIEW_OFF <b>Value0x964</b>
FCodes.KBD_CONFIGURE_TC_COLOR	KBD_CONFIGURE_TC_COLOR <b>Value0x990</b>
FCodes.KBD_TC_USER_NAME	KBD_TC_USER_NAME <b>Value0x991</b>
FCodes.KBD_TC_SEARCH_DOC	KBD_TC_SEARCH_DOC <b>Value0x133</b>
FCodes.KBD_TC_SEARCH_BOOK	KBD_TC_SEARCH_BOOK <b>Value0x134</b>
FCodes.KBD_TC_SEARCH_DITAMAP	KBD_TC_SEARCH_DITAMAP <b>Value0x135</b>

Property name	Data Type
FCodes.KBD_TC_SEARCH_SEL	KBD_TC_SEARCH_SEL <b>Value0x136</b>
FCodes.KBD_TOC	KBD_TOC <b>Value0x2B0</b>
FCodes.KBD_LIST_FIGURE	KBD_LIST_FIGURE <b>Value0x2B1</b>
FCodes.KBD_LIST_TABLE	KBD_LIST_TABLE <b>Value0x2B2</b>
FCodes.KBD_LIST_PGF	KBD_LIST_PGF <b>Value0x2B3</b>
FCodes.KBD_LIST_PGF_ALPHA	KBD_LIST_PGF_ALPHA <b>Value0x2B4</b>
FCodes.KBD_LIST_MARKER	KBD_LIST_MARKER <b>Value0x2B5</b>
FCodes.KBD_LIST_MARKER_ALPHA	KBD_LIST_MARKER_ALPHA <b>Value0x2B6</b>
FCodes.KBD_LIST_REFERENCES	KBD_LIST_REFERENCES <b>Value0x2B7</b>
FCodes.KBD_INDEX_STANDARD	KBD_INDEX_STANDARD <b>Value0x2C0</b>
FCodes.KBD_INDEX_AUTHOR	KBD_INDEX_AUTHOR <b>Value0x2C1</b>
FCodes.KBD_INDEX_SUBJECT	KBD_INDEX_SUBJECT <b>Value0x2C2</b>
FCodes.KBD_INDEX_MARKER	KBD_INDEX_MARKER <b>Value0x2C3</b>
FCodes.KBD_INDEX_REFERENCES	KBD_INDEX_REFERENCES <b>Value0x2C4</b>
FCodes.KBD_OPTIONS	KBD_OPTIONS <b>Value0x360</b>
FCodes.KBD_BORDERS	KBD_BORDERS <b>Value0x361</b>
FCodes.KBD_SYMBOLS	KBD_SYMBOLS <b>Value0x362</b>

Property name	Data Type
FCodes.KBD_RULERS	KBD_RULERS <b>Value0x363</b>
FCodes.KBD_GRID	KBD_GRID <b>Value0x364</b>
FCodes.KBD_SEPARATIONS	KBD_SEPARATIONS <b>Value0x365</b>
FCodes.KBD_COLOR	KBD_COLOR <b>Value0x39F</b>
FCodes.KBD_CONDVISIBILITY	KBD_CONDVISIBILITY <b>Value0x367</b>
FCodes.KBD_CONDTOGGLEOVERR	KBD_CONDTOGGLEOVERR <b>Value0x368</b>
FCodes.KBD_HOTSPOTINDICATORS	KBD_HOTSPOTINDICATORS <b>Value0x508</b>
FCodes.KBD_TOGGLEDRAW	KBD_TOGGLEDRAW <b>Value0x366</b>
FCodes.KBD_VIEWSEP1	KBD_VIEWSEP1 <b>Value0x36D</b>
FCodes.KBD_VIEWSEP2	KBD_VIEWSEP2 <b>Value0x36E</b>
FCodes.KBD_VIEWSEP3	KBD_VIEWSEP3 <b>Value0x36F</b>
FCodes.KBD_VIEWSEP4	KBD_VIEWSEP4 <b>Value0x37D</b>
FCodes.KBD_VIEWSEP5	KBD_VIEWSEP5 <b>Value0x37E</b>
FCodes.KBD_VIEWSEP6	KBD_VIEWSEP6 <b>Value0x39E</b>
FCodes.KBD_MENUCOMPLETE	KBD_MENUCOMPLETE <b>Value0x33D</b>
FCodes.KBD_MENUQUICK	KBD_MENUQUICK <b>Value0x33E</b>
FCodes.KBD_MENUCUSTOM	KBD_MENUCUSTOM <b>Value0x33F</b>

Property name	Data Type
FCodes.KBD_MENU MODIFY	KBD_MENU MODIFY <b>Value0x3D9</b>
FCodes.KBD_MENU RESET	KBD_MENU RESET <b>Value0x3D8</b>
FCodes.KBD_SHOW_BORDERS	KBD_SHOW_BORDERS <b>Value0x3F1</b>
FCodes.KBD_SHOW_HOTSPOT INDICATORS	KBD_SHOW_HOTSPOT INDICATORS <b>Value0x509</b>
FCodes.KBD_SHOW_SYMBOLS	KBD_SHOW_SYMBOLS <b>Value0x3F2</b>
FCodes.KBD_SHOW_RULERS	KBD_SHOW_RULERS <b>Value0x3F3</b>
FCodes.KBD_SHOW_GRID	KBD_SHOW_GRID <b>Value0x3F4</b>
FCodes.KBD_SHOW_LINK_BOUNDARIES	KBD_SHOW_LINK_BOUNDARIES <b>Value0x3F5</b>
FCodes.KBD_SHOW_ELEM_BORDER	KBD_SHOW_ELEM_BORDER <b>Value0x3F6</b>
FCodes.KBD_SHOW_ELEM_TAGS	KBD_SHOW_ELEM_TAGS <b>Value0x3F7</b>
FCodes.KBD_SHOW_GRAPHICS	KBD_SHOW_GRAPHICS <b>Value0x3F8</b>
FCodes.KBD_SHOW_COND_IND	KBD_SHOW_COND_IND <b>Value0x3F9</b>
FCodes.KBD_HIDE_BORDERS	KBD_HIDE_BORDERS <b>Value0x4F1</b>
FCodes.KBD_HIDE_HOTSPOT INDICATORS	KBD_HIDE_HOTSPOT INDICATORS <b>Value0x50A</b>
FCodes.KBD_HIDE_SYMBOLS	KBD_HIDE_SYMBOLS <b>Value0x4F2</b>
FCodes.KBD_HIDE_RULERS	KBD_HIDE_RULERS <b>Value0x4F3</b>
FCodes.KBD_HIDE_GRID	KBD_HIDE_GRID <b>Value0x4F4</b>

Property name	Data Type
FCodes.KBD_HIDE_LINK_BOUNDARIES	KBD_HIDE_LINK_BOUNDARIES <b>Value0x4F5</b>
FCodes.KBD_HIDE_ELEM_BORDER	KBD_HIDE_ELEM_BORDER <b>Value0x4F6</b>
FCodes.KBD_HIDE_GRAPHICS	KBD_HIDE_GRAPHICS <b>Value0x4F8</b>
FCodes.KBD_HIDE_COND_IND	KBD_HIDE_COND_IND <b>Value0x4F9</b>
FCodes.KBD_INSETPOD	KBD_INSETPOD <b>Value0x994</b>
FCodes.KBD_SEARCHREFS	KBD_SEARCHREFS <b>Value0xB17</b>
FCodes.KBD_FLIPUD	KBD_FLIPUD <b>Value0x370</b>
FCodes.KBD_FLIPLR	KBD_FLIPLR <b>Value0x371</b>
FCodes.KBD_ROTATE_CCW	KBD_ROTATE_CCW <b>Value0x372</b>
FCodes.KBD_ROT_PLUS	KBD_ROT_PLUS <b>Value0x372</b>
FCodes.KBD_ROTATE_CCW_SMALL	KBD_ROTATE_CCW_SMALL <b>Value0x38A</b>
FCodes.KBD_SCALE	KBD_SCALE <b>Value0x373</b>
FCodes.KBD_SMOOTH	KBD_SMOOTH <b>Value0x374</b>
FCodes.KBD_UNSMOOTH	KBD_UNSMOOTH <b>Value0x375</b>
FCodes.KBD_RESHAPE	KBD_RESHAPE <b>Value0x376</b>
FCodes.KBD_JOINCURVES	KBD_JOINCURVES <b>Value0x377</b>
FCodes.KBD_SETSIDES	KBD_SETSIDES <b>Value0x378</b>

Property name	Data Type
FCodes.KBD_CONSTRAIN	KBD_CONSTRAIN <b>Value0x379</b>
FCodes.KBD_SNAP	KBD_SNAP <b>Value0x37A</b>
FCodes.KBD_GRAVITY	KBD_GRAVITY <b>Value0x37B</b>
FCodes.KBD_KEEPTOOL	KBD_KEEPTOOL <b>Value0x37C</b>
FCodes.KBD_OVERPRINT	KBD_OVERPRINT <b>Value0x3D2</b>
FCodes.KBD_OVERPRINT_NONE	KBD_OVERPRINT_NONE <b>Value0x3D4</b>
FCodes.KBD_RUN_OFF	KBD_RUN_OFF <b>Value0x670</b>
FCodes.KBD_RUN_CONTOUR	KBD_RUN_CONTOUR <b>Value0x668</b>
FCodes.KBD_RUN_BBOX	KBD_RUN_BBOX <b>Value0x669</b>
FCodes.KBD_RUN_GAP	KBD_RUN_GAP <b>Value0x671</b>
FCodes.KBD_RUN_PROPS	KBD_RUN_PROPS <b>Value0x667</b>
FCodes.KBD_SETRUN_PROPS	KBD_SETRUN_PROPS <b>Value0x96C</b>
FCodes.KBD_FRONT	KBD_FRONT <b>Value0x380</b>
FCodes.KBD_BACK	KBD_BACK <b>Value0x381</b>
FCodes.KBD_GROUP	KBD_GROUP <b>Value0x382</b>
FCodes.KBD_UNGROUP	KBD_UNGROUP <b>Value0x383</b>
FCodes.KBD_ALIGN	KBD_ALIGN <b>Value0x384</b>

Property name	Data Type
FCodes.KBD_DISTRIBUTE	KBD_DISTRIBUTE <b>Value0x385</b>
FCodes.KBD_ROTATE_CW	KBD_ROTATE_CW <b>Value0x386</b>
FCodes.KBD_ROT_MINUS	KBD_ROT_MINUS <b>Value0x386</b>
FCodes.KBD_ROTATE_CW_SMALL	KBD_ROTATE_CW_SMALL <b>Value0x389</b>
FCodes.KBD_OBJPROPS	KBD_OBJPROPS <b>Value0x387</b>
FCodes.KBD_PICKOBJPROPS	KBD_PICKOBJPROPS <b>Value0x388</b>
FCodes.KBD_MUTATE	KBD_MUTATE <b>Value0x37F</b>
FCodes.KBD_ROTATE	KBD_ROTATE <b>Value0x38B</b>
FCodes.KBD_ROTATE_AGAIN	KBD_ROTATE_AGAIN <b>Value0x38C</b>
FCodes.KBD_ROTATE_NATURAL	KBD_ROTATE_NATURAL <b>Value0x38D</b>
FCodes.KBD_REROTATE	KBD_REROTATE <b>Value0x38E</b>
FCodes.KBD_ROTATE_RESETO	KBD_ROTATE_RESETO <b>Value0x38F</b>
FCodes.KBD_SETALIGN_PROPS	KBD_SETALIGN_PROPS <b>Value0x992</b>
FCodes.KBD_SETDISTRIBUTE_PROPS	KBD_SETDISTRIBUTE_PROPS <b>Value0x993</b>
FCodes.KBD_CLOSEWIN	KBD_CLOSEWIN <b>Value0x390</b>
FCodes.KBD_OPENWIN	KBD_OPENWIN <b>Value0x391</b>
FCodes.KBD_CLOPWIN	KBD_CLOPWIN <b>Value0x392</b>

Property name	Data Type
FCodes.KBD_MOVEWIN	KBD_MOVEWIN <b>Value0x393</b>
FCodes.KBD_EXPOSEWIN	KBD_EXPOSEWIN <b>Value0x395</b>
FCodes.KBD_HIDEWIN	KBD_HIDEWIN <b>Value0x396</b>
FCodes.KBD_HISHWIN	KBD_HISHWIN <b>Value0x397</b>
FCodes.KBD_REFRESHWIN	KBD_REFRESHWIN <b>Value0x398</b>
FCodes.KBD_QUITWIN	KBD_QUITWIN <b>Value0x399</b>
FCodes.KBD_RESIZELOCK	KBD_RESIZELOCK <b>Value0x39A</b>
FCodes.KBD_RESIZEUNLOCK	KBD_RESIZEUNLOCK <b>Value0x39B</b>
FCodes.KBD_QUIETCLOSEWIN	KBD_QUIETCLOSEWIN <b>Value0x39C</b>
FCodes.KBD_POPUP_MENU	KBD_POPUP_MENU <b>Value0x39D</b>
FCodes.KBD_LINETOOL	KBD_LINETOOL <b>Value0x3A0</b>
FCodes.KBD_RECTTOOL	KBD_RECTTOOL <b>Value0x3A1</b>
FCodes.KBD_POLYGTOOL	KBD_POLYGTOOL <b>Value0x3A2</b>
FCodes.KBD_POLYLTOOL	KBD_POLYLTOOL <b>Value0x3A3</b>
FCodes.KBD_ARCTOOL	KBD_ARCTOOL <b>Value0x3A4</b>
FCodes.KBD_ROUNDRECT	KBD_ROUNDRECT <b>Value0x3A5</b>
FCodes.KBD_OVALTOOL	KBD_OVALTOOL <b>Value0x3A6</b>



Property name	Data Type
FCodes.KBD_TEXTLTOOL	KBD_TEXTLTOOL <b>Value0x3A7</b>
FCodes.KBD_TEXTRTOOL	KBD_TEXTRTOOL <b>Value0x3A8</b>
FCodes.KBD_FREETOOL	KBD_FREETOOL <b>Value0x3A9</b>
FCodes.KBD_FRAMETOOL	KBD_FRAMETOOL <b>Value0x3AA</b>
FCodes.KBD_LASTTOOL	KBD_LASTTOOL <b>Value0x3AB</b>
FCodes.KBD_WIDTH0	KBD_WIDTH0 <b>Value0x3AC</b>
FCodes.KBD_WIDTH1	KBD_WIDTH1 <b>Value0x3AD</b>
FCodes.KBD_INCWIDTH	KBD_INCWIDTH <b>Value0x3AE</b>
FCodes.KBD_DECWIDTH	KBD_DECWIDTH <b>Value0x3AF</b>
FCodes.KBD_PEN0	KBD_PEN0 <b>Value0x3B0</b>
FCodes.KBD_PEN1	KBD_PEN1 <b>Value0x3B1</b>
FCodes.KBD_INCPEN	KBD_INCPEN <b>Value0x3B2</b>
FCodes.KBD_DECPEN	KBD_DECPEN <b>Value0x3B3</b>
FCodes.KBD_FILL0	KBD_FILL0 <b>Value0x3B4</b>
FCodes.KBD_FILL1	KBD_FILL1 <b>Value0x3B5</b>
FCodes.KBD_INCFILL	KBD_INCFILL <b>Value0x3B6</b>
FCodes.KBD_DECFILL	KBD_DECFILL <b>Value0x3B7</b>

Property name	Data Type
FCodes.KBD_SETFILL	KBD_SETFILL <b>Value0x3B8</b>
FCodes.KBD_SETPEN	KBD_SETPEN <b>Value0x3B9</b>
FCodes.KBD_SETWIDTH	KBD_SETWIDTH <b>Value0x3BA</b>
FCodes.KBD_SETCAP	KBD_SETCAP <b>Value0x3BB</b>
FCodes.KBD_SETSEP	KBD_SETSEP <b>Value0x3BC</b>
FCodes.KBD_SETSEP_KEEP	KBD_SETSEP_KEEP <b>Value0x3BE</b>
FCodes.KBD_SETSEP_ALL	KBD_SETSEP_ALL <b>Value0x3BF</b>
FCodes.KBD_SETSEP_RESET_TINT_OVERPRINT	KBD_SETSEP_RESET_TINT_OVERPRINT <b>Value0x3D1</b>
FCodes.KBD_SETFILL_0	KBD_SETFILL_0 <b>Value0x430</b>
FCodes.KBD_SETFILL_1	KBD_SETFILL_1 <b>Value0x431</b>
FCodes.KBD_SETFILL_2	KBD_SETFILL_2 <b>Value0x432</b>
FCodes.KBD_SETFILL_3	KBD_SETFILL_3 <b>Value0x433</b>
FCodes.KBD_SETFILL_4	KBD_SETFILL_4 <b>Value0x434</b>
FCodes.KBD_SETFILL_5	KBD_SETFILL_5 <b>Value0x435</b>
FCodes.KBD_SETFILL_6	KBD_SETFILL_6 <b>Value0x436</b>
FCodes.KBD_SETFILL_7	KBD_SETFILL_7 <b>Value0x437</b>
FCodes.KBD_SETFILL_8	KBD_SETFILL_8 <b>Value0x438</b>

Property name	Data Type
FCodes.KBD_SETFILL_9	KBD_SETFILL_9 <b>Value0x439</b>
FCodes.KBD_SETFILL_A	KBD_SETFILL_A <b>Value0x43A</b>
FCodes.KBD_SETFILL_B	KBD_SETFILL_B <b>Value0x43B</b>
FCodes.KBD_SETFILL_C	KBD_SETFILL_C <b>Value0x43C</b>
FCodes.KBD_SETFILL_D	KBD_SETFILL_D <b>Value0x43D</b>
FCodes.KBD_SETFILL_E	KBD_SETFILL_E <b>Value0x43E</b>
FCodes.KBD_SETFILL_F	KBD_SETFILL_F <b>Value0x43F</b>
FCodes.KBD_SETPEN_0	KBD_SETPEN_0 <b>Value0x440</b>
FCodes.KBD_SETPEN_1	KBD_SETPEN_1 <b>Value0x441</b>
FCodes.KBD_SETPEN_2	KBD_SETPEN_2 <b>Value0x442</b>
FCodes.KBD_SETPEN_3	KBD_SETPEN_3 <b>Value0x443</b>
FCodes.KBD_SETPEN_4	KBD_SETPEN_4 <b>Value0x444</b>
FCodes.KBD_SETPEN_5	KBD_SETPEN_5 <b>Value0x445</b>
FCodes.KBD_SETPEN_6	KBD_SETPEN_6 <b>Value0x446</b>
FCodes.KBD_SETPEN_7	KBD_SETPEN_7 <b>Value0x447</b>
FCodes.KBD_SETPEN_8	KBD_SETPEN_8 <b>Value0x448</b>
FCodes.KBD_SETPEN_9	KBD_SETPEN_9 <b>Value0x449</b>

Property name	Data Type
FCodes.KBD_SETPEN_A	KBD_SETPEN_A <b>Value0x44A</b>
FCodes.KBD_SETPEN_B	KBD_SETPEN_B <b>Value0x44B</b>
FCodes.KBD_SETPEN_C	KBD_SETPEN_C <b>Value0x44C</b>
FCodes.KBD_SETPEN_D	KBD_SETPEN_D <b>Value0x44D</b>
FCodes.KBD_SETPEN_E	KBD_SETPEN_E <b>Value0x44E</b>
FCodes.KBD_SETPEN_F	KBD_SETPEN_F <b>Value0x44F</b>
FCodes.KBD_SETWIDTH_0	KBD_SETWIDTH_0 <b>Value0x450</b>
FCodes.KBD_SETWIDTH_1	KBD_SETWIDTH_1 <b>Value0x451</b>
FCodes.KBD_SETWIDTH_2	KBD_SETWIDTH_2 <b>Value0x452</b>
FCodes.KBD_SETWIDTH_3	KBD_SETWIDTH_3 <b>Value0x453</b>
FCodes.KBD_SETWIDTH_SLIDE	KBD_SETWIDTH_SLIDE <b>Value0x45E</b>
FCodes.KBD_SETWIDTH_OPTION	KBD_SETWIDTH_OPTION <b>Value0x45F</b>
FCodes.KBD_SETCAP_0	KBD_SETCAP_0 <b>Value0x460</b>
FCodes.KBD_SETCAP_1	KBD_SETCAP_1 <b>Value0x461</b>
FCodes.KBD_SETCAP_2	KBD_SETCAP_2 <b>Value0x462</b>
FCodes.KBD_SETCAP_3	KBD_SETCAP_3 <b>Value0x463</b>
FCodes.KBD_SETCAP_OPTION	KBD_SETCAP_OPTION <b>Value0x46F</b>

Property name	Data Type
FCodes.KBD_SETDASH_0	KBD_SETDASH_0 <b>Value0x470</b>
FCodes.KBD_SETDASH_1	KBD_SETDASH_1 <b>Value0x471</b>
FCodes.KBD_SETDASH_2	KBD_SETDASH_2 <b>Value0x472</b>
FCodes.KBD_SETDASH_3	KBD_SETDASH_3 <b>Value0x473</b>
FCodes.KBD_SETDASH_4	KBD_SETDASH_4 <b>Value0x474</b>
FCodes.KBD_SETDASH_5	KBD_SETDASH_5 <b>Value0x475</b>
FCodes.KBD_SETDASH_6	KBD_SETDASH_6 <b>Value0x476</b>
FCodes.KBD_SETDASH_7	KBD_SETDASH_7 <b>Value0x477</b>
FCodes.KBD_SETDASH_8	KBD_SETDASH_8 <b>Value0x478</b>
FCodes.KBD_SETDASH_OPTION	KBD_SETDASH_OPTION <b>Value0x47F</b>
FCodes.KBD_SNAP_0	KBD_SNAP_0 <b>Value0x480</b>
FCodes.KBD_SNAP_1	KBD_SNAP_1 <b>Value0x481</b>
FCodes.KBD_GRAVITY_0	KBD_GRAVITY_0 <b>Value0x482</b>
FCodes.KBD_GRAVITY_1	KBD_GRAVITY_1 <b>Value0x483</b>
FCodes.KBD_SETKNOCKOUT	KBD_SETKNOCKOUT <b>Value0x48A</b>
FCodes.KBD_SETOVERPRINT	KBD_SETOVERPRINT <b>Value0x48B</b>
FCodes.KBD_SETFROMCOLOR	KBD_SETFROMCOLOR <b>Value0x48C</b>

Property name	Data Type
FCodes.KBD_SETTINT	KBD_SETTINT <b>Value0x48D</b>
FCodes.KBD_SETSOLID	KBD_SETSOLID <b>Value0x402</b>
FCodes.KBD_SETDASH	KBD_SETDASH <b>Value0x403</b>
FCodes.KBD_DASH0	KBD_DASH0 <b>Value0x404</b>
FCodes.KBD_DASH1	KBD_DASH1 <b>Value0x405</b>
FCodes.KBD_INCDASH	KBD_INCDASH <b>Value0x406</b>
FCodes.KBD_DECDASH	KBD_DECDASH <b>Value0x407</b>
FCodes.KBD_DASHOPTION	KBD_DASHOPTION <b>Value0x408</b>
FCodes.KBD_CHECKSEL	KBD_CHECKSEL <b>Value0x3C0</b>
FCodes.KBD_CHECKDOC	KBD_CHECKDOC <b>Value0x3C1</b>
FCodes.KBD_CORRECT	KBD_CORRECT <b>Value0x3C2</b>
FCodes.KBD_ADDUSRDICTION	KBD_ADDUSRDICTION <b>Value0x3C3</b>
FCodes.KBD_ADDDOCDICTION	KBD_ADDDOCDICTION <b>Value0x3C4</b>
FCodes.KBD_ADDAUTOCORR	KBD_ADDAUTOCORR <b>Value0x3C5</b>
FCodes.KBD_DELUSRDICTION	KBD_DELUSRDICTION <b>Value0x3C6</b>
FCodes.KBD_DELDOCDICTION	KBD_DELDOCDICTION <b>Value0x3C7</b>
FCodes.KBD_CLEARAUTO	KBD_CLEARAUTO <b>Value0x3C8</b>

Property name	Data Type
FCodes.KBD_CHANGEDICT	KBD_CHANGEDICT <b>Value0x3C9</b>
FCodes.KBD_SPELLRESET	KBD_SPELLRESET <b>Value0x3CA</b>
FCodes.KBD_CHECKPAGE	KBD_CHECKPAGE <b>Value0x3CB</b>
FCodes.KBD_SOPTIONS	KBD_SOPTIONS <b>Value0x3CC</b>
FCodes.KBD_HYPHENATE	KBD_HYPHENATE <b>Value0x3CD</b>
FCodes.KBD_CHECKBATCH	KBD_CHECKBATCH <b>Value0x3CE</b>
FCodes.KBD_REFORMATDOC	KBD_REFORMATDOC <b>Value0x3CF</b>
FCodes.KBD_ASC_CORRECT	KBD_ASC_CORRECT <b>Value0x995</b>
FCodes.KBD_ASC_ADDUSRDICTIONARY	KBD_ASC_ADDUSRDICTIONARY <b>Value0x996</b>
FCodes.KBD_ASC_ADDDOCDICTIONARY	KBD_ASC_ADDDOCDICTIONARY <b>Value0x997</b>
FCodes.KBD_ASC_ENABLE_AUTO_SPELL_CHECK	KBD_ASC_ENABLE_AUTO_SPELL_CHECK <b>Value0x999</b>
FCodes.TEXT_INSET_PROPS	TEXT_INSET_PROPS <b>Value0x3E0</b>
FCodes.UPDATE_INSETS	UPDATE_INSETS <b>Value0x3E1</b>
FCodes.SUPPRESS_INSETS	SUPPRESS_INSETS <b>Value0x3E2</b>
FCodes.KBD_CREATE_MULTIMEDIA_LINK	KBD_CREATE_MULTIMEDIA_LINK <b>Value0x3E3</b>
FCodes.KBD_CREATE_MULTIMEDIA_LINK_TBL	KBD_CREATE_MULTIMEDIA_LINK_TBL <b>Value0x3E4</b>
FCodes.KBD_MENUBARFOCUS	KBD_MENUBARFOCUS <b>Value0x3F0</b>

Property name	Data Type
FCodes.KBD_SINGLE_QUOTE	KBD_SINGLE_QUOTE <b>Value0x400</b>
FCodes.KBD_DOUBLE_QUOTE	KBD_DOUBLE_QUOTE <b>Value0x401</b>
FCodes.HIGH_CHAR_PREV	HIGH_CHAR_PREV <b>Value0x410</b>
FCodes.HIGH_WORD_PREV	HIGH_WORD_PREV <b>Value0x411</b>
FCodes.HIGH_LINE_PREV	HIGH_LINE_PREV <b>Value0x412</b>
FCodes.HIGH_SENT_PREV	HIGH_SENT_PREV <b>Value0x413</b>
FCodes.HIGH_PGF_PREV	HIGH_PGF_PREV <b>Value0x414</b>
FCodes.HIGH_LINE_UP	HIGH_LINE_UP <b>Value0x415</b>
FCodes.HIGH_LINE_DOWN	HIGH_LINE_DOWN <b>Value0x416</b>
FCodes.HIGH_COL_TOP	HIGH_COL_TOP <b>Value0x417</b>
FCodes.HIGH_COL_BOT	HIGH_COL_BOT <b>Value0x418</b>
FCodes.HIGH_FLOW_BEG	HIGH_FLOW_BEG <b>Value0x419</b>
FCodes.HIGH_FLOW_END	HIGH_FLOW_END <b>Value0x41A</b>
FCodes.HIGH_LINE_BEG	HIGH_LINE_BEG <b>Value0x41B</b>
FCodes.HIGH_LINE_END	HIGH_LINE_END <b>Value0x41C</b>
FCodes.HIGH_ELEMENT_PREV	HIGH_ELEMENT_PREV <b>Value0x420</b>
FCodes.HIGH_ELEMENT_NEXT	HIGH_ELEMENT_NEXT <b>Value0x421</b>



Property name	Data Type
FCodes.HIGH_SIBLINGS	HIGH_SIBLINGS <b>Value0x422</b>
FCodes.HIGH_PARENT	HIGH_PARENT <b>Value0x423</b>
FCodes.KBD_DFN_WEB_obsolete	KBD_DFN_WEB_obsolete <b>Value0x500</b>
FCodes.KBD_DFN_LIB_obsolete	KBD_DFN_LIB_obsolete <b>Value0x501</b>
FCodes.KBD_PASTE_RTF	KBD_PASTE_RTF <b>Value0x502</b>
FCodes.KBD_PAGELAYOUT	KBD_PAGELAYOUT <b>Value0x503</b>
FCodes.KBD_COPYRIGHT	KBD_COPYRIGHT <b>Value0x504</b>
FCodes.KBD_RESTOREFONT	KBD_RESTOREFONT <b>Value0x505</b>
FCodes.KBD_SYMFONT	KBD_SYMFONT <b>Value0x506</b>
FCodes.HIGH_CHAR_NEXT	HIGH_CHAR_NEXT <b>Value0x510</b>
FCodes.HIGH_WORD_NEXT	HIGH_WORD_NEXT <b>Value0x511</b>
FCodes.HIGH_LINE_NEXT	HIGH_LINE_NEXT <b>Value0x512</b>
FCodes.HIGH_SENT_NEXT	HIGH_SENT_NEXT <b>Value0x513</b>
FCodes.HIGH_PGF_NEXT	HIGH_PGF_NEXT <b>Value0x514</b>
FCodes.HIGH_SAMECB	HIGH_SAMECB <b>Value0x515</b>
FCodes.HIGH_SAMECOND	HIGH_SAMECOND <b>Value0x516</b>
FCodes.HIGH_HYPertext	HIGH_HYPertext <b>Value0x517</b>

Property name	Data Type
FCodes.SEARCH_CB	SEARCH_CB <b>Value0x518</b>
FCodes.SEARCH_CBTAG	SEARCH_CBTAG <b>Value0x519</b>
FCodes.SEARCH_COND	SEARCH_COND <b>Value0x51A</b>
FCodes.SEARCH_CBOVERRIDE	SEARCH_CBOVERRIDE <b>Value0x51B</b>
FCodes.OBJ_SEL_FIRST	OBJ_SEL_FIRST <b>Value0x600</b>
FCodes.OBJ_SEL_NEXT	OBJ_SEL_NEXT <b>Value0x601</b>
FCodes.OBJ_SEL_EXTEND_NEXT	OBJ_SEL_EXTEND_NEXT <b>Value0x602</b>
FCodes.OBJ_SEL_PREV	OBJ_SEL_PREV <b>Value0x603</b>
FCodes.OBJ_SEL_NEXT_WRAP	OBJ_SEL_NEXT_WRAP <b>Value0x604</b>
FCodes.OBJ_SEL_PREV_WRAP	OBJ_SEL_PREV_WRAP <b>Value0x605</b>
FCodes.OBJ_SEL_LAST	OBJ_SEL_LAST <b>Value0x606</b>
FCodes.FOCUS_INPUT_DOC	FOCUS_INPUT_DOC <b>Value0x620</b>
FCodes.FOCUS_INPUT_SEARCH	FOCUS_INPUT_SEARCH <b>Value0x621</b>
FCodes.FOCUS_INPUT_MARKER	FOCUS_INPUT_MARKER <b>Value0x622</b>
FCodes.FOCUS_INPUT_SPELL	FOCUS_INPUT_SPELL <b>Value0x623</b>
FCodes.FOCUS_INPUT_HYPertext	FOCUS_INPUT_HYPertext <b>Value0x624</b>
FCodes.FOCUS_INPUT_PGFFMT	FOCUS_INPUT_PGFFMT <b>Value0x626</b>

Property name	Data Type
FCodes.FOCUS_INPUT_FONTFMT	FOCUS_INPUT_FONTFMT <b>Value0x627</b>
FCodes.FOCUS_INPUT_COND	FOCUS_INPUT_COND <b>Value0x629</b>
FCodes.FOCUS_INPUT_CELLFMT	FOCUS_INPUT_CELLFMT <b>Value0x62A</b>
FCodes.FOCUS_INPUT_CUSTRS	FOCUS_INPUT_CUSTRS <b>Value0x62B</b>
FCodes.FOCUS_INPUT_TBLFMT	FOCUS_INPUT_TBLFMT <b>Value0x62C</b>
FCodes.FOCUS_INPUT_STRWIN	FOCUS_INPUT_STRWIN <b>Value0x62D</b>
FCodes.FOCUS_INPUT_VALIDATION	FOCUS_INPUT_VALIDATION <b>Value0x62E</b>
FCodes.FOCUS_INPUT_ATTREDITOR	FOCUS_INPUT_ATTREDITOR <b>Value0x62F</b>
FCodes.FOCUS_INPUT_ELEM_CTX	FOCUS_INPUT_ELEM_CTX <b>Value0xA32</b>
FCodes.KBD_CLOSE_SEARCH	KBD_CLOSE_SEARCH <b>Value0x681</b>
FCodes.KBD_CLOSE_MARKER	KBD_CLOSE_MARKER <b>Value0x682</b>
FCodes.KBD_CLOSE_SPELL	KBD_CLOSE_SPELL <b>Value0x683</b>
FCodes.KBD_CLOSE_PGFFMT	KBD_CLOSE_PGFFMT <b>Value0x684</b>
FCodes.KBD_CLOSE_FONTFMT	KBD_CLOSE_FONTFMT <b>Value0x685</b>
FCodes.KBD_CLOSE_COND	KBD_CLOSE_COND <b>Value0x686</b>
FCodes.KBD_CLOSE_CUSTRS	KBD_CLOSE_CUSTRS <b>Value0x687</b>
FCodes.KBD_CLOSE_TBLFMT	KBD_CLOSE_TBLFMT <b>Value0x688</b>

Property name	Data Type
FCodes.KBD_CLOSE_STRWIN	KBD_CLOSE_STRWIN <b>Value0x689</b>
FCodes.KBD_CLOSE_VALIDATION	KBD_CLOSE_VALIDATION <b>Value0x68A</b>
FCodes.KBD_CLOSE_ATTREDITOR	KBD_CLOSE_ATTREDITOR <b>Value0x68B</b>
FCodes.KBD_CLOSE_ELEM_CTX	KBD_CLOSE_ELEM_CTX <b>Value0x68C</b>
FCodes.KBD_CLOSE_HYPertext	KBD_CLOSE_HYPertext <b>Value0x68D</b>
FCodes.KBD_PAUSE_1x00	KBD_PAUSE_1x00 <b>Value0x630</b>
FCodes.KBD_PAUSE_0x10	KBD_PAUSE_0x10 <b>Value0x631</b>
FCodes.KBD_PAUSE_0x01	KBD_PAUSE_0x01 <b>Value0x632</b>
FCodes.KBD_SLOW_0x00	KBD_SLOW_0x00 <b>Value0x633</b>
FCodes.KBD_SLOW_1x00	KBD_SLOW_1x00 <b>Value0x634</b>
FCodes.KBD_SLOW_0x10	KBD_SLOW_0x10 <b>Value0x635</b>
FCodes.KBD_SLOW_0x01	KBD_SLOW_0x01 <b>Value0x636</b>
FCodes.KBD_SET_TEXTFRAME_GRID	KBD_SET_TEXTFRAME_GRID <b>Value0x666</b>
FCodes.KBD_PURIFY_NEW_LEAKS	KBD_PURIFY_NEW_LEAKS <b>Value0x650</b>
FCodes.KBD_PURIFY_ALL_LEAKS	KBD_PURIFY_ALL_LEAKS <b>Value0x651</b>
FCodes.KBD_PURIFY_CLEAR_LEAKS	KBD_PURIFY_CLEAR_LEAKS <b>Value0x652</b>
FCodes.KBD_COUNT_SBLOCKS	KBD_COUNT_SBLOCKS <b>Value0x653</b>

Property name	Data Type
FCodes.KBD_API	KBD_API <b>Value0xDF0</b>
FCodes.KBD_API_SHORTCUT	KBD_API_SHORTCUT <b>Value0xDF1</b>
FCodes.TYPEIN	TYPEIN <b>Value0xDF2</b>
FCodes.INLINE_TYPEIN	INLINE_TYPEIN <b>Value0xDF3</b>
FCodes.KBD_VIEW_API	KBD_VIEW_API <b>Value0xDF4</b>
FCodes.KBD_VIEW_API_SHORTCUT	KBD_VIEW_API_SHORTCUT <b>Value0xDF5</b>
FCodes.KBD_VIEW_SWITCH	KBD_VIEW_SWITCH <b>Value0xDF6</b>
FCodes.KBD_BACKSTACK	KBD_BACKSTACK <b>Value0xEF0</b>
FCodes.KBD_UPSTACK	KBD_UPSTACK <b>Value0xEF1</b>
FCodes.KBD_VIEWER	KBD_VIEWER <b>Value0xF00</b>
FCodes.TOGGLE_FLUID_VIEW	TOGGLE_FLUID_VIEW <b>Value0x500</b>
FCodes.KBD_DSEXIT	KBD_DSEXIT <b>Value0xF01</b>
FCodes.KBD_MEMFAIL	KBD_MEMFAIL <b>Value0xF02</b>
FCodes.KBD_SAVEMETA	KBD_SAVEMETA <b>Value0xF03</b>
FCodes.KBD_MEM_STATS	KBD_MEM_STATS <b>Value0xF04</b>
FCodes.KBD_CACHE_STATS	KBD_CACHE_STATS <b>Value0xF05</b>
FCodes.KBD_NEWVAR	KBD_NEWVAR <b>Value0xF06</b>

Property name	Data Type
FCodes.KBD_UPDATEREF	KBD_UPDATEREF <b>Value0xF07</b>
FCodes.KBD_DEREFREF	KBD_DEREFREF <b>Value0xF08</b>
FCodes.KBD_HEATREF	KBD_HEATREF <b>Value0xF09</b>
FCodes.KBD_DOC_REPORT	KBD_DOC_REPORT <b>Value0xF10</b>
FCodes.KBD_FULLRULERS	KBD_FULLRULERS <b>Value0x3BD</b>
FCodes.KBD_ALLSELECT	KBD_ALLSELECT <b>Value0xF20</b>
FCodes.KBD_OBJSELECT	KBD_OBJSELECT <b>Value0xF21</b>
FCodes.KBD_TOOLWIN	KBD_TOOLWIN <b>Value0xF22</b>
FCodes.KBD_PGFWIN	KBD_PGFWIN <b>Value0xF23</b>
FCodes.KBD_FONTWIN	KBD_FONTWIN <b>Value0xF24</b>
FCodes.KBD_RESIZEBOX	KBD_RESIZEBOX <b>Value0xF25</b>
FCodes.KBD_MATHWIN	KBD_MATHWIN <b>Value0xF26</b>
FCodes.KBD_RESIZEBOXM	KBD_RESIZEBOXM <b>Value0xF27</b>
FCodes.KBD_ELEMENTWIN	KBD_ELEMENTWIN <b>Value0xF28</b>
FCodes.KBD_SMALLTOOLWIN	KBD_SMALLTOOLWIN <b>Value0xF29</b>
FCodes.KBD_OBJSELECT_NOPREF	KBD_OBJSELECT_NOPREF <b>Value0xF2A</b>
FCodes.KBD_HOTSPOT	KBD_HOTSPOT <b>Value0x4FD</b>

Property name	Data Type
FCodes.KBD_EDITHOTSPOT	KBD_EDITHOTSPOT <b>Value0x4FE</b>
FCodes.KBD_DELINKHOTSPOT	KBD_DELINKHOTSPOT <b>Value0x507</b>
FCodes.KBD_ZOOMIN	KBD_ZOOMIN <b>Value0xF30</b>
FCodes.KBD_ZOOMOUT	KBD_ZOOMOUT <b>Value0xF31</b>
FCodes.KBD_ZOOM_FIT_PAGE	KBD_ZOOM_FIT_PAGE <b>Value0xF32</b>
FCodes.KBD_ZOOM_FIT_WINDOW	KBD_ZOOM_FIT_WINDOW <b>Value0xF33</b>
FCodes.KBD_ZOOM	KBD_ZOOM <b>Value0xF34</b>
FCodes.KBD_ZOOM100	KBD_ZOOM100 <b>Value0xF35</b>
FCodes.KBD_ZOOM_SET	KBD_ZOOM_SET <b>Value0xF3C</b>
FCodes.KBD_ZOOM_FIT_TEXTFRAME	KBD_ZOOM_FIT_TEXTFRAME <b>Value0xF3F</b>
FCodes.KBD_ZOOM_AUTOFIT_AFTER_ZOOM	KBD_ZOOM_AUTOFIT_AFTER_ZOOM <b>Value0xF1A</b>
FCodes.KBD_TAGSTATUS	KBD_TAGSTATUS <b>Value0xF36</b>
FCodes.KBD_PAGESTATUS	KBD_PAGESTATUS <b>Value0xF37</b>
FCodes.KBD_HSCROLL	KBD_HSCROLL <b>Value0xF38</b>
FCodes.KBD_VSCROLL	KBD_VSCROLL <b>Value0xF39</b>
FCodes.KBD_RENAMEPAGE	KBD_RENAMEPAGE <b>Value0xF3A</b>
FCodes.KBD_RENAMEFRAME	KBD_RENAMEFRAME <b>Value0xF3B</b>

Property name	Data Type
FCodes.KBD_FONTQUICK	KBD_FONTQUICK <b>Value0xF40</b>
FCodes.KBD_PGFQUICK	KBD_PGFQUICK <b>Value0xF41</b>
FCodes.KBD_VARQUICK	KBD_VARQUICK <b>Value0xF42</b>
FCodes.KBD_CELLFMTQUICK	KBD_CELLFMTQUICK <b>Value0xF43</b>
FCodes.KBD_CONDINQUICK	KBD_CONDINQUICK <b>Value0xF44</b>
FCodes.KBD_CONDNOTINQUICK	KBD_CONDNOTINQUICK <b>Value0xF45</b>
FCodes.KBD_UNCOND	KBD_UNCOND <b>Value0xF46</b>
FCodes.KBD_CONDVISONLYQUICK	KBD_CONDVISONLYQUICK <b>Value0xF47</b>
FCodes.KBD_INSERTQUICK	KBD_INSERTQUICK <b>Value0xF48</b>
FCodes.KBD_WRAPQUICK	KBD_WRAPQUICK <b>Value0xF49</b>
FCodes.KBD_CHANGEQUICK	KBD_CHANGEQUICK <b>Value0xF4C</b>
FCodes.KBD_ATTREDITQUICK	KBD_ATTREDITQUICK <b>Value0xF4D</b>
FCodes.PGF_DESIGN_CAT	PGF_DESIGN_CAT <b>Value0xF4E</b>
FCodes.CHAR_DESIGN_CAT	CHAR_DESIGN_CAT <b>Value0xF4F</b>
FCodes.TBL_DESIGN_CAT	TBL_DESIGN_CAT <b>Value0xF5F</b>
FCodes.KBD_NOCHANGEDB	KBD_NOCHANGEDB <b>Value0xF4A</b>
FCodes.KBD_RESETDB	KBD_RESETDB <b>Value0xF4B</b>



Property name	Data Type
FCodes.KBD_SMEQN	KBD_SMEQN <b>Value0xF50</b>
FCodes.KBD_MEDEQN	KBD_MEDEQN <b>Value0xF51</b>
FCodes.KBD_LGEQN	KBD_LGEQN <b>Value0xF52</b>
FCodes.KBD_PUTINLINE	KBD_PUTINLINE <b>Value0xF53</b>
FCodes.KBD_ANTIPUTINLINE	KBD_ANTIPUTINLINE <b>Value0xF54</b>
FCodes.KBD_EVACUATE	KBD_EVACUATE <b>Value0xF60</b>
FCodes.KBD_VERIFYCONTEXT	KBD_VERIFYCONTEXT <b>Value0xF61</b>
FCodes.KBD_SAVEASDBRE	KBD_SAVEASDBRE <b>Value0xF70</b>
FCodes.KBD_TEST_MODAL	KBD_TEST_MODAL <b>Value0xF71</b>
FCodes.KBD_TEST_MODELESS	KBD_TEST_MODELESS <b>Value0xF72</b>
FCodes.KBD_STUFF_ITEM	KBD_STUFF_ITEM <b>Value0xF73</b>
FCodes.KBD_SAVESAS	KBD_SAVESAS <b>Value0xF74</b>
FCodes.KBD_TEST_PRINTDBRE	KBD_TEST_PRINTDBRE <b>Value0xF75</b>
FCodes.KBD_SAVEFMX	KBD_SAVEFMX <b>Value0xF76</b>
FCodes.KBD_SAVEDBRE	KBD_SAVEDBRE <b>Value0xF77</b>
FCodes.KBD_CAPTURE_LIVE_DIALOG	KBD_CAPTURE_LIVE_DIALOG <b>Value0xF78</b>
FCodes.KBD_RM_MODE	KBD_RM_MODE <b>Value0xF79</b>

Property name	Data Type
FCodes.KBD_DRE_MODE_X	KBD_DRE_MODE_X <b>Value0xF7A</b>
FCodes.KBD_DRE_MODE_W	KBD_DRE_MODE_W <b>Value0xF7B</b>
FCodes.KBD_DRE_MODE_M	KBD_DRE_MODE_M <b>Value0xF7C</b>
FCodes.KBD_TABLE_INS	KBD_TABLE_INS <b>Value0xF80</b>
FCodes.KBD_TABLE_FORMAT	KBD_TABLE_FORMAT <b>Value0xF81</b>
FCodes.KBD_TABLE_CELLFMT	KBD_TABLE_CELLFMT <b>Value0xF84</b>
FCodes.KBD_TABLE_ROWFMFMT	KBD_TABLE_ROWFMFMT <b>Value0xF85</b>
FCodes.KBD_TABLE_CUSTRS	KBD_TABLE_CUSTRS <b>Value0xF86</b>
FCodes.KBD_TABLE_ADDRC	KBD_TABLE_ADDRC <b>Value0xF87</b>
FCodes.KBD_TABLE_RESIZECOL	KBD_TABLE_RESIZECOL <b>Value0xF88</b>
FCodes.KBD_TABLE_STRADDLE	KBD_TABLE_STRADDLE <b>Value0xF89</b>
FCodes.KBD_TABLE_CONVERT	KBD_TABLE_CONVERT <b>Value0xF8A</b>
FCodes.KBD_TABLE_DEBUG	KBD_TABLE_DEBUG <b>Value0xF8B</b>
FCodes.KBD_TABLE_RULES	KBD_TABLE_RULES <b>Value0xF8C</b>
FCodes.KBD_TABLE_EXIT_IP	KBD_TABLE_EXIT_IP <b>Value0xFC0</b>
FCodes.KBD_TBL_DLG_UNIFY_TF	KBD_TBL_DLG_UNIFY_TF <b>Value0xF90</b>
FCodes.KBD_TBL_DLG_UNIFY_CF	KBD_TBL_DLG_UNIFY_CF <b>Value0xF91</b>

Property name	Data Type
FCodes.KBD_TBL_DLG_ADD_ABOVE	KBD_TBL_DLG_ADD_ABOVE <b>Value0xF92</b>
FCodes.KBD_TBL_DLG_ADD_BELOW	KBD_TBL_DLG_ADD_BELOW <b>Value0xF93</b>
FCodes.KBD_TBL_DLG_ADD_LEFT	KBD_TBL_DLG_ADD_LEFT <b>Value0xF94</b>
FCodes.KBD_TBL_DLG_ADD_RIGHT	KBD_TBL_DLG_ADD_RIGHT <b>Value0xF95</b>
FCodes.KBD_TBL_DLG_CLEAR_EMPTY	KBD_TBL_DLG_CLEAR_EMPTY <b>Value0xF96</b>
FCodes.KBD_TBL_DLG_CLEAR_X	KBD_TBL_DLG_CLEAR_X <b>Value0xF97</b>
FCodes.KBD_TBL_DLG_PASTE_REPL	KBD_TBL_DLG_PASTE_REPL <b>Value0xF98</b>
FCodes.KBD_TBL_DLG_PASTE_BEFORE	KBD_TBL_DLG_PASTE_BEFORE <b>Value0xF99</b>
FCodes.KBD_TBL_DLG_PASTE_AFTER	KBD_TBL_DLG_PASTE_AFTER <b>Value0xF9A</b>
FCodes.KBD_TABLEWIN	KBD_TABLEWIN <b>Value0xFBB</b>
FCodes.TBL_APPLY_CAT_TO_SEL	TBL_APPLY_CAT_TO_SEL <b>Value0xFBC</b>
FCodes.KBD_CLOSE_TABLE_CATALOG	KBD_CLOSE_TABLE_CATALOG <b>Value0xFBD</b>
FCodes.TBL_APPLY_TAG	TBL_APPLY_TAG <b>Value0xFBE</b>
FCodes.KBD_COPYCELLFMT	KBD_COPYCELLFMT <b>Value0xF9B</b>
FCodes.KBD_COPYCOLW	KBD_COPYCOLW <b>Value0xF9C</b>
FCodes.KBD_COPYCOND	KBD_COPYCOND <b>Value0xF9D</b>
FCodes.KBD_USEELTDEFSFROM	KBD_USEELTDEFSFROM <b>Value0xF9E</b>

Property name	Data Type
FCodes.KBD_COPYATTRS	KBD_COPYATTRS <b>Value0xF9F</b>
FCodes.KBD_ATOMIZE_INSET	KBD_ATOMIZE_INSET <b>Value0xFAA</b>
FCodes.KBD_SWAP_RED_BLUE	KBD_SWAP_RED_BLUE <b>Value0xFAB</b>
FCodes.KBD_REWRAP_INLINE_MATH	KBD_REWRAP_INLINE_MATH <b>Value0xFAC</b>
FCodes.KBD_MODE_ROTATE_TOOL	KBD_MODE_ROTATE_TOOL <b>Value0xFAD</b>
FCodes.KBD_XYZZY	KBD_XYZZY <b>Value0xFED</b>
FCodes.KBD_XYZZZ	KBD_XYZZZ <b>Value0xEE0</b>
FCodes.KBD_XYZZQ	KBD_XYZZQ <b>Value0xEE1</b>
FCodes.KBD_ACTIVATE_IN_PDF_BYDEFAULT	KBD_ACTIVATE_IN_PDF_BYDEFAULT <b>Value0x4FA</b>
FCodes.KBD_OPEN_IN_POPUP_WIN_IN_PDF	KBD_OPEN_IN_POPUP_WIN_IN_PDF <b>Value0x4FB</b>
FCodes.KBD_GRAPHIC_SET_NAME	KBD_GRAPHIC_SET_NAME <b>Value0x4FC</b>
FCodes.CMD_OPEN_DITAMAP_COMPONENT	CMD_OPEN_DITAMAP_COMPONENT <b>Value0xF14</b>
FCodes.PGF_DESIGNKIT_APPLY	PGF_DESIGNKIT_APPLY <b>Value0xAAA</b>
FCodes.CHAR_DESIGNKIT_APPLY	CHAR_DESIGNKIT_APPLY <b>Value0xAAB</b>
FCodes.TBL_DESIGNKIT_APPLY	TBL_DESIGNKIT_APPLY <b>Value0xAAC</b>
FCodes.PGF_DESIGNKIT_UPDATEALL	PGF_DESIGNKIT_UPDATEALL <b>Value0xAAD</b>
FCodes.CHAR_DESIGNKIT_UPDATEALL	CHAR_DESIGNKIT_UPDATEALL <b>Value0xAAE</b>

Property name	Data Type
FCodes.TBL_DESIGNKIT_UPDATEALL	TBL_DESIGNKIT_UPDATEALL <b>Value0xAAF</b>
FCodes.PGF_DESIGNKIT_NEWFORMAT	PGF_DESIGNKIT_NEWFORMAT <b>Value0xB00</b>
FCodes.CHAR_DESIGNKIT_NEWFORMAT	CHAR_DESIGNKIT_NEWFORMAT <b>Value0xB01</b>
FCodes.TBL_DESIGNKIT_NEWFORMAT	TBL_DESIGNKIT_NEWFORMAT <b>Value0xB02</b>
FCodes.PGF_DESIGNKIT_UPDATEOPTION S	PGF_DESIGNKIT_UPDATEOPTIONS <b>Value0xB03</b>
FCodes.CHAR_DESIGNKIT_UPDATEOPTIO NS	CHAR_DESIGNKIT_UPDATEOPTIONS <b>Value0xB04</b>
FCodes.TBL_DESIGNKIT_UPDATEOPTION S	TBL_DESIGNKIT_UPDATEOPTIONS <b>Value0xB05</b>
FCodes.KBD_PGFFMT_DELETE	KBD_PGFFMT_DELETE <b>Value0xB06</b>
FCodes.KBD_CHARFMT_DELETE	KBD_CHARFMT_DELETE <b>Value0xB07</b>
FCodes.KBD_TBLFMT_DELETE	KBD_TBLFMT_DELETE <b>Value0xB08</b>
FCodes.KBD_STYLEFMT_DELETE_UNUSE D	KBD_STYLEFMT_DELETE_UNUSED <b>Value0xB10</b>
FCodes.KBD_PGFFMT_DELETE_UNUSED	KBD_PGFFMT_DELETE_UNUSED <b>Value0xB13</b>
FCodes.KBD_CHARFMT_DELETE_UNUSE D	KBD_CHARFMT_DELETE_UNUSED <b>Value0xB14</b>
FCodes.KBD_TBLFMT_DELETE_UNUSED	KBD_TBLFMT_DELETE_UNUSED <b>Value0xB15</b>
FCodes.PGF_DESIGNKIT_RESET	PGF_DESIGNKIT_RESET <b>Value0xABA</b>
FCodes.CHAR_DESIGNKIT_RESET	CHAR_DESIGNKIT_RESET <b>Value0xABB</b>
FCodes.TBL_DESIGNKIT_RESET	TBL_DESIGNKIT_RESET <b>Value0xABC</b>

Property name	Data Type
FCodes.KBD_STYLEFMT_DELETE	KBD_STYLEFMT_DELETE <b>Value0xBB0</b>
FCodes.KBD_STYLE_CATALOG	KBD_STYLE_CATALOG <b>Value0xBB1</b>
FCodes.KBD_STYLE_DESIGNER	KBD_STYLE_DESIGNER <b>Value0xBB2</b>
FCodes.CONDTEXT_KIT_APPLY	CONDTEXT_KIT_APPLY <b>Value0xB09</b>
FCodes.ALT_TXT_INSET	ALT_TXT_INSET <b>Value0xB11</b>
FCodes.ALT_GFX_INSET	ALT_GFX_INSET <b>Value0xB12</b>
FCodes.KBD_TBLSEL_CELL	KBD_TBLSEL_CELL <b>Value0xFA0</b>
FCodes.KBD_TBLSEL_ROW	KBD_TBLSEL_ROW <b>Value0xFA1</b>
FCodes.KBD_TBLSEL_COL	KBD_TBLSEL_COL <b>Value0xFA2</b>
FCodes.KBD_TBLSEL_TABLE	KBD_TBLSEL_TABLE <b>Value0xFA3</b>
FCodes.KBD_TBLSEL_CELLTEXT	KBD_TBLSEL_CELLTEXT <b>Value0xFA4</b>
FCodes.KBD_TBLSEL_COLBODY	KBD_TBLSEL_COLBODY <b>Value0xFA5</b>
FCodes.KBD_TBLIP_TOPLEFT	KBD_TBLIP_TOPLEFT <b>Value0xFB0</b>
FCodes.KBD_TBLIP_RIGHT	KBD_TBLIP_RIGHT <b>Value0xFB1</b>
FCodes.KBD_TBLIP_LEFT	KBD_TBLIP_LEFT <b>Value0xFB2</b>
FCodes.KBD_TBLIP_ABOVE	KBD_TBLIP_ABOVE <b>Value0xFB3</b>
FCodes.KBD_TBLIP_BELOW	KBD_TBLIP_BELOW <b>Value0xFB4</b>

Property name	Data Type
FCodes.KBD_TBLIP_LEFTMOST	KBD_TBLIP_LEFTMOST <b>Value0xFB5</b>
FCodes.KBD_TBLIP_RIGHTMOST	KBD_TBLIP_RIGHTMOST <b>Value0xFB6</b>
FCodes.KBD_TBLIP_NEXT	KBD_TBLIP_NEXT <b>Value0xFB7</b>
FCodes.KBD_TBLIP_PREV	KBD_TBLIP_PREV <b>Value0xFB8</b>
FCodes.KBD_TBLIP_TOP	KBD_TBLIP_TOP <b>Value0xFB9</b>
FCodes.KBD_TBLIP_BOTTOM	KBD_TBLIP_BOTTOM <b>Value0xFBA</b>
FCodes.PGF_KBD_SIDEBODY	PGF_KBD_SIDEBODY <b>Value0xFC9</b>
FCodes.PGF_KBD_SIDEHEAD_FIRST_BASELINE	PGF_KBD_SIDEHEAD_FIRST_BASELINE <b>Value0xFCA</b>
FCodes.PGF_KBD_SIDEHEAD_LAST_BASELINE	PGF_KBD_SIDEHEAD_LAST_BASELINE <b>Value0xFCB</b>
FCodes.PGF_KBD_SIDEHEAD_TOP	PGF_KBD_SIDEHEAD_TOP <b>Value0xFCC</b>
FCodes.PGF_KBD_RUN_IN	PGF_KBD_RUN_IN <b>Value0xFCD</b>
FCodes.PGF_KBD_FULLSTRADDLE	PGF_KBD_FULLSTRADDLE <b>Value0xFCE</b>
FCodes.PGF_KBD_BODYSTRADDLE	PGF_KBD_BODYSTRADDLE <b>Value0xFCF</b>
FCodes.KBD_TBL_DLG_SHRINKWRAP	KBD_TBL_DLG_SHRINKWRAP <b>Value0xFD0</b>
FCodes.ELEM_INS_CAT_AT_SEL	ELEM_INS_CAT_AT_SEL <b>Value0xA10</b>
FCodes.ELEM_WRAP_CAT_AT_SEL	ELEM_WRAP_CAT_AT_SEL <b>Value0xA11</b>
FCodes.ELEM_CHANGE_CAT_AT_SEL	ELEM_CHANGE_CAT_AT_SEL <b>Value0xA12</b>

Property name	Data Type
FCodes.ELEM_INSERT_CUSTOM_1	ELEM_INSERT_CUSTOM_1 <b>Value0xA21</b>
FCodes.ELEM_INSERT_CUSTOM_2	ELEM_INSERT_CUSTOM_2 <b>Value0xA22</b>
FCodes.ELEM_INSERT_CUSTOM_3	ELEM_INSERT_CUSTOM_3 <b>Value0xA23</b>
FCodes.ELEM_INSERT_CUSTOM_4	ELEM_INSERT_CUSTOM_4 <b>Value0xA24</b>
FCodes.ELEM_INSERT_CUSTOM_5	ELEM_INSERT_CUSTOM_5 <b>Value0xA25</b>
FCodes.ELEM_INSERT_CUSTOM_6	ELEM_INSERT_CUSTOM_6 <b>Value0xA26</b>
FCodes.ELEM_INSERT_CUSTOM_7	ELEM_INSERT_CUSTOM_7 <b>Value0xA27</b>
FCodes.ELEM_INSERT_CUSTOM_8	ELEM_INSERT_CUSTOM_8 <b>Value0xA28</b>
FCodes.ELEM_INSERT_CUSTOM_9	ELEM_INSERT_CUSTOM_9 <b>Value0xA29</b>
FCodes.ELEM_INSERT_CUSTOM_10	ELEM_INSERT_CUSTOM_10 <b>Value0xA2A</b>
FCodes.KBD_STRIP_FLOWSTRUCTURE	KBD_STRIP_FLOWSTRUCTURE <b>Value0xA30</b>
FCodes.KBD_SHOW_ELEM_CTX	KBD_SHOW_ELEM_CTX <b>Value0xA31</b>
FCodes.KBD_NORMALIZE_TAGS	KBD_NORMALIZE_TAGS <b>Value0xA33</b>
FCodes.KBD_STRWIN	KBD_STRWIN <b>Value0xFD1</b>
FCodes.KBD_ELEM_BORDER	KBD_ELEM_BORDER <b>Value0xFD2</b>
FCodes.KBD_ELEM_MERGE_1ST	KBD_ELEM_MERGE_1ST <b>Value0xFD3</b>
FCodes.KBD_ELEM_MERGE_LAST	KBD_ELEM_MERGE_LAST <b>Value0xFD4</b>



Property name	Data Type
FCodes.KBD_ELEM_SPLIT	KBD_ELEM_SPLIT <b>Value0xFD5</b>
FCodes.KBD_ELEM_UNWRAP	KBD_ELEM_UNWRAP <b>Value0xFD6</b>
FCodes.KBD_ELEM_CAT_OPTS	KBD_ELEM_CAT_OPTS <b>Value0xFD7</b>
FCodes.KBD_SETELCATSTRICT	KBD_SETELCATSTRICT <b>Value0xFD8</b>
FCodes.KBD_SETELCATLOOSE	KBD_SETELCATLOOSE <b>Value0xFD9</b>
FCodes.KBD_SETELCATCHILD	KBD_SETELCATCHILD <b>Value0xFDA</b>
FCodes.KBD_SETELCATALL	KBD_SETELCATALL <b>Value0xFDB</b>
FCodes.KBD_SETELCATFREQ	KBD_SETELCATFREQ <b>Value0xFF7</b>
FCodes.ELEM_BAM	ELEM_BAM <b>Value0xFDC</b>
FCodes.KBD_ELEM_TAGS	KBD_ELEM_TAGS <b>Value0xFDD</b>
FCodes.KBD_BANNER_TEXT	KBD_BANNER_TEXT <b>Value0x137</b>
FCodes.KBD_ELEM_PROMOTE	KBD_ELEM_PROMOTE <b>Value0xFDE</b>
FCodes.KBD_ELEM_DEMOTE	KBD_ELEM_DEMOTE <b>Value0xFDF</b>
FCodes.KBD_VALIDATION	KBD_VALIDATION <b>Value0xFE0</b>
FCodes.KBD_VAL_ELEM	KBD_VAL_ELEM <b>Value0xFE1</b>
FCodes.KBD_VAL_FLOW	KBD_VAL_FLOW <b>Value0xFE2</b>
FCodes.KBD_VAL_DOC	KBD_VAL_DOC <b>Value0xFE3</b>

Property name	Data Type
FCodes.KBD_VAL_IGNORE	KBD_VAL_IGNORE <b>Value0xFE4</b>
FCodes.KBD_VAL_START	KBD_VAL_START <b>Value0xFE5</b>
FCodes.KBD_VAL_ALLOW	KBD_VAL_ALLOW <b>Value0xFE6</b>
FCodes.KBD_VAL_CLEAR	KBD_VAL_CLEAR <b>Value0xFE7</b>
FCodes.SW_CSR_UP	SW_CSR_UP <b>Value0xFE8</b>
FCodes.SW_CSR_DOWN	SW_CSR_DOWN <b>Value0xFE9</b>
FCodes.SW_CSR_RIGHT	SW_CSR_RIGHT <b>Value0xFEa</b>
FCodes.SW_CSR_LEFT	SW_CSR_LEFT <b>Value0xFEB</b>
FCodes.KBD_INLINE_ATTREDTR	KBD_INLINE_ATTREDTR <b>Value0x29B</b>
FCodes.KBD_ATTRIBUTE_EDIT	KBD_ATTRIBUTE_EDIT <b>Value0xFEC</b>
FCodes.KBD_ATTR_CONFIG_FILE_MAKER	KBD_ATTR_CONFIG_FILE_MAKER <b>Value0xFFA</b>
FCodes.KBD_ATTR_DISP_OPTS	KBD_ATTR_DISP_OPTS <b>Value0xFEE</b>
FCodes.KBD_FB_EDIT_PREFS	KBD_FB_EDIT_PREFS <b>Value0xFEF</b>
FCodes.KBD_REMOVE_STRUCT	KBD_REMOVE_STRUCT <b>Value0xFF0</b>
FCodes.KBD_TOGGLE_INCLUSION	KBD_TOGGLE_INCLUSION <b>Value0xFF1</b>
FCodes.KBD_TOGGLE_COLLAPSE	KBD_TOGGLE_COLLAPSE <b>Value0xFF2</b>
FCodes.KBD_TOGGLE_COLLAPSE_ALL	KBD_TOGGLE_COLLAPSE_ALL <b>Value0xFF3</b>

Property name	Data Type
FCodes.KBD_TOGGLE_COLLAPSE_CHILDREN	KBD_TOGGLE_COLLAPSE_CHILDREN <b>Value0xA0D</b>
FCodes.KBD_ELEM_TRANSPOSE_PREV	KBD_ELEM_TRANSPOSE_PREV <b>Value0xFF4</b>
FCodes.KBD_ELEM_TRANSPOSE_NEXT	KBD_ELEM_TRANSPOSE_NEXT <b>Value0xFF5</b>
FCodes.KBD_VAL_IGNORE_ATTR	KBD_VAL_IGNORE_ATTR <b>Value0xFF6</b>
FCodes.KBD_NAMESPACES	KBD_NAMESPACES <b>Value0xFF8</b>
FCodes.KBD_TOGGLE_STRUCT_AND_DOCWIN	KBD_TOGGLE_STRUCT_AND_DOCWIN <b>Value0xFF9</b>
FCodes.FM_SEPARATOR	FM_SEPARATOR <b>Value0xF12</b>
FCodes.FM_TERMINATE	FM_TERMINATE <b>Value0xFFF</b>
FCodes.FM_RPT_CMDS_BY_TAG	FM_RPT_CMDS_BY_TAG <b>Value0xD00</b>
FCodes.FM_RPT_CMDS_BY_SHORTCUT	FM_RPT_CMDS_BY_SHORTCUT <b>Value0xD01</b>
FCodes.KBD_PASTESPECIAL	KBD_PASTESPECIAL <b>Value0xD10</b>
FCodes.KBD_CLOSE_TOP_PALETTE	KBD_CLOSE_TOP_PALETTE <b>Value0xD20</b>
FCodes.KBD_CLOSE_PARAGRAPH_CATALOG	KBD_CLOSE_PARAGRAPH_CATALOG <b>Value0xD21</b>
FCodes.KBD_CLOSE_CHARACTER_CATALOG	KBD_CLOSE_CHARACTER_CATALOG <b>Value0xD22</b>
FCodes.KBD_CLOSE_TOOLS_PALETTE	KBD_CLOSE_TOOLS_PALETTE <b>Value0xD23</b>
FCodes.KBD_CLOSE_EQUATIONS_PALETTE	KBD_CLOSE_EQUATIONS_PALETTE <b>Value0xD24</b>
FCodes.KBD_CLOSE_ELEMENT_CATALOG	KBD_CLOSE_ELEMENT_CATALOG <b>Value0xD26</b>

Property name	Data Type
FCodes.KBD_CLOSE_PARAGRAPH_DESIGNER	KBD_CLOSE_PARAGRAPH_DESIGNER <b>Value0xD28</b>
FCodes.KBD_CLOSE_CHARACTER_DESIGNER	KBD_CLOSE_CHARACTER_DESIGNER <b>Value0xD29</b>
FCodes.KBD_CLOSE_TABLE_DESIGNER	KBD_CLOSE_TABLE_DESIGNER <b>Value0xD2A</b>
FCodes.KBD_WINDOWFULL_UP	KBD_WINDOWFULL_UP <b>Value0xD40</b>
FCodes.KBD_WINDOWFULL_DOWN	KBD_WINDOWFULL_DOWN <b>Value0xD41</b>
FCodes.KBD_VIEWER_MAKE_DOCUMENT	KBD_VIEWER_MAKE_DOCUMENT <b>Value0xD50</b>
FCodes.KBD_VIEWER_MAKE_PALETTE	KBD_VIEWER_MAKE_PALETTE <b>Value0xD51</b>
FCodes.KBD_VIEWER_MAKE_MODELESSDIALOG	KBD_VIEWER_MAKE_MODELESSDIALOG <b>Value0xD52</b>
FCodes.KBD_Item1stLogical	KBD_Item1stLogical <b>Value0xD60</b>
FCodes.KBD_ItemNextLogical	KBD_ItemNextLogical <b>Value0xD61</b>
FCodes.KBD_ItemPrevLogical	KBD_ItemPrevLogical <b>Value0xD62</b>
FCodes.KBD_ItemNextPhysical	KBD_ItemNextPhysical <b>Value0xD63</b>
FCodes.KBD_ItemPrevPhysical	KBD_ItemPrevPhysical <b>Value0xD64</b>
FCodes.KBD_ItemLeft	KBD_ItemLeft <b>Value0xD65</b>
FCodes.KBD_ItemRight	KBD_ItemRight <b>Value0xD66</b>
FCodes.KBD_ItemUp	KBD_ItemUp <b>Value0xD67</b>
FCodes.KBD_ItemDown	KBD_ItemDown <b>Value0xD68</b>

Property name	Data Type
FCodes.KBD_ItemSelect	KBD_ItemSelect <b>Value0xD69</b>
FCodes.KBD_ItemFirstFocus	KBD_ItemFirstFocus <b>Value0xD6A</b>
FCodes.INIT_DIALOG	INIT_DIALOG <b>Value0xD80</b>
FCodes.KBD_RENAMEORPLAIN	KBD_RENAMEORPLAIN <b>Value0x900</b>
FCodes.KBD_WIN_CASCADE	KBD_WIN_CASCADE <b>Value0x903</b>
FCodes.KBD_WIN_TILE	KBD_WIN_TILE <b>Value0x904</b>
FCodes.KBD_TAB_LEFT	KBD_TAB_LEFT <b>Value0x906</b>
FCodes.KBD_TAB_CENTER	KBD_TAB_CENTER <b>Value0x907</b>
FCodes.KBD_TAB_RIGHT	KBD_TAB_RIGHT <b>Value0x908</b>
FCodes.KBD_TAB_DECIMAL	KBD_TAB_DECIMAL <b>Value0x909</b>
FCodes.KBD_PRINTSETUP	KBD_PRINTSETUP <b>Value0x90A</b>
FCodes.KBD_MINIMIZE	KBD_MINIMIZE <b>Value0x90F</b>
FCodes.KBD_MAXIMIZE	KBD_MAXIMIZE <b>Value0x910</b>
FCodes.KBD_RESTORE	KBD_RESTORE <b>Value0x911</b>
FCodes.KBD_EDITLINKS	KBD_EDITLINKS <b>Value0x912</b>
FCodes.KBD_INSERTOBJECT	KBD_INSERTOBJECT <b>Value0x913</b>
FCodes.KBD_CONTEXTMENU	KBD_CONTEXTMENU <b>Value0x914</b>

Property name	Data Type
FCodes.KBD_EMBEDDED00	KBD_EMBEDDED00 <b>Value0x920</b>
FCodes.KBD_EMBEDDED01	KBD_EMBEDDED01 <b>Value0x921</b>
FCodes.KBD_EMBEDDED02	KBD_EMBEDDED02 <b>Value0x922</b>
FCodes.KBD_EMBEDDED03	KBD_EMBEDDED03 <b>Value0x923</b>
FCodes.KBD_EMBEDDED04	KBD_EMBEDDED04 <b>Value0x924</b>
FCodes.KBD_EMBEDDED05	KBD_EMBEDDED05 <b>Value0x925</b>
FCodes.KBD_EMBEDDED06	KBD_EMBEDDED06 <b>Value0x926</b>
FCodes.KBD_EMBEDDED07	KBD_EMBEDDED07 <b>Value0x927</b>
FCodes.KBD_EMBEDDED08	KBD_EMBEDDED08 <b>Value0x928</b>
FCodes.KBD_EMBEDDED09	KBD_EMBEDDED09 <b>Value0x92A</b>
FCodes.KBD_EMBEDDED10	KBD_EMBEDDED10 <b>Value0x92B</b>
FCodes.KBD_EMBEDDED11	KBD_EMBEDDED11 <b>Value0x92C</b>
FCodes.KBD_EMBEDDED12	KBD_EMBEDDED12 <b>Value0x92D</b>
FCodes.KBD_EMBEDDED13	KBD_EMBEDDED13 <b>Value0x92E</b>
FCodes.KBD_EMBEDDED14	KBD_EMBEDDED14 <b>Value0x92F</b>
FCodes.KBD_EMBEDDED15	KBD_EMBEDDED15 <b>Value0x930</b>
FCodes.KBD_EMBEDDED16	KBD_EMBEDDED16 <b>Value0x931</b>

Property name	Data Type
FCodes.KBD_EMBEDDED17	KBD_EMBEDDED17 <b>Value0x932</b>
FCodes.KBD_EMBEDDED18	KBD_EMBEDDED18 <b>Value0x933</b>
FCodes.KBD_EMBEDDED19	KBD_EMBEDDED19 <b>Value0x934</b>
FCodes.KBD_EMBEDDEDMAX	KBD_EMBEDDEDMAX <b>Value0x94F</b>
FCodes.KBD_GOTOLINEN	KBD_GOTOLINEN <b>Value0x970</b>
FCodes.KBD_SHOWNEXT	KBD_SHOWNEXT <b>Value0x971</b>
FCodes.KBD_SHOWPREV	KBD_SHOWPREV <b>Value0x972</b>
FCodes.KBD_SCREENMODE_TOGGLE	KBD_SCREENMODE_TOGGLE <b>Value0x978</b>
FCodes.KBD_SCREENMODE_STANDARD	KBD_SCREENMODE_STANDARD <b>Value0x979</b>
FCodes.KBD_SCREENMODE_FULLSCREEN_UI	KBD_SCREENMODE_FULLSCREEN_UI <b>Value0x97A</b>
FCodes.KBD_SCREENMODE_FULLSCREEN	KBD_SCREENMODE_FULLSCREEN <b>Value0x97B</b>
FCodes.KBD_UI_PREFERENCE	KBD_UI_PREFERENCE <b>Value0x980</b>
FCodes.KBD_UIALERTSTRINGS_PREF	KBD_UIALERTSTRINGS_PREF <b>Value0x981</b>
FCodes.KBD_TOOLBAR_SHOWALL	KBD_TOOLBAR_SHOWALL <b>Value0x989</b>
FCodes.KBD_TOOLBAR_HIDEALL	KBD_TOOLBAR_HIDEALL <b>Value0x98A</b>
FCodes.KBD_TOOLBAR_BASE	KBD_TOOLBAR_BASE <b>Value0x98F</b>
FCodes.KBD_BROWSE_URL	KBD_BROWSE_URL <b>Value0x9A0</b>

Property name	Data Type
FCodes.KBD_MATH_BASE	KBD_MATH_BASE <b>Value0x1000</b>
FCodes.TEXTSEL_QUICK_COPY	TEXTSEL_QUICK_COPY <b>Value0x0001</b>
FCodes.TEXTSEL_EXTEND	TEXTSEL_EXTEND <b>Value0x0002</b>
FCodes.TEXTSEL_EXTEND_WORD	TEXTSEL_EXTEND_WORD <b>Value0x0004</b>
FCodes.TEXTSEL_EXTEND_LINE	TEXTSEL_EXTEND_LINE <b>Value0x0008</b>
FCodes.TEXTSEL_EXTEND_PGF	TEXTSEL_EXTEND_PGF <b>Value0x0010</b>
FCodes.TEXTSEL_WORD	TEXTSEL_WORD <b>Value0x0020</b>
FCodes.TEXTSEL_LINE	TEXTSEL_LINE <b>Value0x0040</b>
FCodes.TEXTSEL_PGF	TEXTSEL_PGF <b>Value0x0080</b>
FCodes.TEXTSEL_SELECT_ONLY	TEXTSEL_SELECT_ONLY <b>Value0x0100</b>
FCodes.TEXTSEL_EXTEND_SENT	TEXTSEL_EXTEND_SENT <b>Value0x0200</b>
FCodes.TEXTSEL_SENT	TEXTSEL_SENT <b>Value0x0400</b>
FCodes.TEXTSEL_EXTEND_ELEMENT	TEXTSEL_EXTEND_ELEMENT <b>Value0x0800</b>
FCodes.TEXTSEL_ELEMENT	TEXTSEL_ELEMENT <b>Value0x1000</b>
FCodes.TEXTSEL_DRAGGING	TEXTSEL_DRAGGING <b>Value0x2000</b>
FCodes.EXTEND_SEL	EXTEND_SEL <b>Value(0x0002   0x0004   0x0008   0x0010   0x0800)</b>
FCodes.MIF_SAVE_TEXT	MIF_SAVE_TEXT <b>Value0x000001</b>



Property name	Data Type
FCodes.MIF_SAVE_TAGS	MIF_SAVE_TAGS <b>Value0x000002</b>
FCodes.MIF_SAVE_FMTS	MIF_SAVE_FMTS <b>Value0x000004</b>
FCodes.MIF_SAVE_FONTS	MIF_SAVE_FONTS <b>Value0x000008</b>
FCodes.MIF_SAVE_MKRS	MIF_SAVE_MKRS <b>Value0x000010</b>
FCodes.MIF_SAVE_AFMS	MIF_SAVE_AFMS <b>Value0x000020</b>
FCodes.MIF_SAVE_LAYT	MIF_SAVE_LAYT <b>Value0x000040</b>
FCodes.MIF_SAVE_MPAGE	MIF_SAVE_MPAGE <b>Value0x000080</b>
FCodes.MIF_SAVE_FCAT	MIF_SAVE_FCAT <b>Value0x000100</b>
FCodes.MIF_SAVE_PCAT	MIF_SAVE_PCAT <b>Value0x000200</b>
FCodes.MIF_SAVE_CCAT	MIF_SAVE_CCAT <b>Value0x000400</b>
FCodes.MIF_SAVE_TMPLT	MIF_SAVE_TMPLT <b>Value0x000800</b>
FCodes.MIF_SAVE_DICT	MIF_SAVE_DICT <b>Value0x001000</b>
FCodes.MIF_SAVE_VARS	MIF_SAVE_VARS <b>Value0x002000</b>
FCodes.MIF_SAVE_TABLECATS	MIF_SAVE_TABLECATS <b>Value0x004000</b>
FCodes.MIF_SAVE_TABLES	MIF_SAVE_TABLES <b>Value0x008000</b>
FCodes.MIF_SAVE_ECAT	MIF_SAVE_ECAT <b>Value0x010000</b>
FCodes.MIF_SAVE_ELEMENTS	MIF_SAVE_ELEMENTS <b>Value0x020000</b>

Property name	Data Type
FCodes.MIF_SAVE_COLORCAT	MIF_SAVE_COLORCAT <b>Value0x040000</b>
FCodes.MIF_SAVE_VIEWSET	MIF_SAVE_VIEWSET <b>Value0x080000</b>
FCodes.MIF_SAVE_DATALINKS	MIF_SAVE_DATALINKS <b>Value0x100000</b>
FCodes.MIF_SAVE_FPLCAT	MIF_SAVE_FPLCAT <b>Value0x200000</b>
FCodes.MIF_SAVE_TEXTINSETS	MIF_SAVE_TEXTINSETS <b>Value0x400000</b>
FCodes.MIF_SAVE_RUBIS	MIF_SAVE_RUBIS <b>Value0x800000</b>
FCodes.MIF_SAVE_BANNERTEXT	MIF_SAVE_BANNERTEXT <b>Value0x1000000</b>
FCodes.KBD_OPEN_REPOSITORY	KBD_OPEN_REPOSITORY <b>Value0x801</b>
FCodes.KBD_CMS_CONNECTION_MGR	KBD_CMS_CONNECTION_MGR <b>Value0x802</b>
FCodes.KBD_NEW_XML	KBD_NEW_XML <b>Value0x803</b>
FCodes.KBD_CMS_UPLOAD_DOC_OR_BOOK	KBD_CMS_UPLOAD_DOC_OR_BOOK <b>Value0x805</b>
FCodes.KBD_REFRESH_DITAMAP_RMVIEW	KBD_REFRESH_DITAMAP_RMVIEW <b>Value0x807</b>
FCodes.STYLE_APPLY_CAT_TO_SEL	STYLE_APPLY_CAT_TO_SEL <b>Value0x808</b>
FCodes.KBD_SHOW_ELEM_CTX_AV	KBD_SHOW_ELEM_CTX_AV <b>Value0x809</b>
FCodes.FOCUS_INPUT_ELEM_CTX_AV	FOCUS_INPUT_ELEM_CTX_AV <b>Value0x810</b>
FCodes.KBD_CLOSE_ELEM_CTX_AV	KBD_CLOSE_ELEM_CTX_AV <b>Value0x811</b>

## FilterArgs

Property name	Data Type	Description
infile	string	The file to filter.
outfile	string	The filter output file.
cname	string	The registered name of the filter
informat	string	The registered incoming format.
outformat	string	The registered output format
binname	string	The name of the filter client file.
args	<i>Strings</i>	Currently NULL.

### FilterArgs methods

[FilterArgs](#).

## Flow

Property name	Data Type	Description
FlowsAutoConnect	int	1 if <code>Autoconnect</code> is enabled; 0 otherwise.
FlowsFeathered	int	1 if <code>Feather</code> is enabled; 0 otherwise
FlowsPostScript	int	1 if flow is PostScript code; 0 otherwise
FlowsSynchronized	int	1 if <code>Baseline Synchronization</code> is enabled; 0 otherwise.
FirstTextFrameInFlow	<i>TextFrame</i>	First text frame in flow (FO_TextFrame ID).
LastTextFrameInFlow	<i>TextFrame</i>	Last text frame in flow (FO_TextFrame ID).
MaxInterlinePadding	int	Maximum interline spacing.
MaxInterPgfPadding	int	Maximum interparagraph spacing.

Property name	Data Type	Description
MinHang	int	Maximum character height for synchronization of first line in column. If characters exceed this height, FrameMaker does not synchronize the first line.
Name	string	Name of flow tag.
NextFlowInDoc	<i>Flow</i>	Next flow in document (FO_Flow ID).
SideHeadRoomInFlow	int	1 if Leave Room for Sideheads in Flow is enabled; 0 otherwise.
Spacing	int	Line spacing for synchronized baselines.
HighestLevelElement	<i>Element</i>	Highest-level element in flow (FO_Element ID)
Direction	int	Set or get the direction of the document. Possible values: <ul style="list-style-type: none"> <li><i>Constants.FV_DIR_Inherit</i> - Inherit the direction of the parent</li> <li><i>Constants.FV_DIR_LTR</i> - Left-to-right</li> <li><i>FV_DIR_RTL</i> - Right-to-left</li> </ul>
ResolvedDirection	int	Get the inherited direction of the document. Possible values: <ul style="list-style-type: none"> <li><i>Constants.FV_DIR_LTR</i> - Left-to-right</li> <li><i>FV_DIR_RTL</i> - Right-to-left</li> </ul>
UserString	string	A string to which clients can store private data.

## Flow methods

*Delete, GetProps, GetText, ObjectValid, SetProps.*

## FmtChangeList

Property name	Data Type	Description
FmtChangeListInCatalog	int	1 if the format change list is in the Format Change List Catalog. 0 if it is in an element definition, as part of the text format rules.
Name	string	The name of the format change list if it is in the Format Change List Catalog.
NextFmtChangeListInDoc	<a href="#">FmtChangeList</a>	The next format change list in the document (FO_FmtChangeList ID).
PgfCatalogReference	string	A paragraph format tag if the format change list specifies one. If this property is set, you cannot change any of the other format change list properties, except Name.
AdjHyphens	int	Number of allowable adjacent hyphens
BottomSeparator	string	Name of frame to put below paragraph.
BottomSepAtIndent	int	1 if the position of the frame specified by BottomSeparator is at the current left indent;0 otherwise.
Hyphenate	int	1 if Automatic Hyphenation is enabled;0, otherwise.
HyphMinPrefix	int	Minimum number of letters that must precede the hyphen.
HyphMinSuffix	int	Minimum number of letters that must follow a hyphen.
HyphMinWord	int	Minimum length of a hyphenated word.
LetterSpace	int	1 if Word Spacing is enabled.
MaxSpace	int	Maximum word spacing (percentage of an em space in current font).
MinSpace	int	Minimum word spacing (percentage of an em space in current font).
OptSpace	int	Optimum word spacing.
TopSeparator	string	Name of frame to put above paragraph.

Property name	Data Type	Description
TopSepAtIndent	int	1 if the position of the frame specified by TopSeparator is at the current left indent; 0 otherwise.
MinJRomSpace	int	Minimum Asian-Roman space.
OptJRomSpace	int	Optimum Asian-Roman space.
MaxJRomSpace	int	Maximum Asian-Roman space.
MinJLetSpace	int	Minimum Asian letter space.
OptJLetSpace	int	Optimum Asian letter space.
MaxJLetSpace	int	Maximum Asian letter space.
YakumonoType	int	Denotes the The Yakumono rules to handle punctuation characters. The value can be one of: <ul style="list-style-type: none"> <li>Constants.FV_FLOATING_YAKUMONO (0x00)</li> <li>Constants.FV_MONOSPACE_YAKUMONO (0x01)</li> <li>Constants.FV_FIXED_YAKUMONO (0x02)</li> </ul>
AutoNumChar	string	Character format for the automatic numbering string specified by AutoNumString; `` ``if the default character format is used.
AutoNumString	string	Autonumber format string (for example, <n>.<n+>)
NumAtEnd	int	1 if the numbering position is at the End of the Paragraph; 0 if it is at the Beginning of the Paragraph.
PgflsAutoNum	int	1 if autonumbering is enabled; 0 otherwise.
FirstIndent	int	The paragraph's first-line left margin, measured from the left side of the current text column (0 cm to 100 cm ).
FirstIndentChange	int	Amount by which to increase or decrease the first-line left margin.

Property name	Data Type	Description
FirstIndentIsRelative	int	1 if the first indent is relative to the left indent.
FirstIndentRelPos	int	Position relative to left indent if <code>FirstIndentIsRelative</code> is True.
Leading	int	Denotes the space below each line in the paragraph.
LeadingChange	int	Amount by which to increase or decrease the leading.
LeftIndent	int	The paragraph's left margin, measured from the left side of the current text column (0 cm to 100 cm).
LeftIndentChange	int	Amount by which to increase or decrease the left margin.
LineSpacingFixed	int	1 if the line spacing is fixed.
MoveTabs	int	Amount by which to move all tab positions in the paragraph.
NumTabs	int	The number of tabs in the paragraph. To clear all the tabs in the paragraph, set <code>NumTabs</code> to 0.
PgfAlignment	int	Horizontal alignment of the paragraph. The value is one of: <ul style="list-style-type: none"> <li>• <code>Constants.FV_PGF_LEFT (0x01)</code> <code>Constants.FV_PGF_RIGHT (0x02)</code> <code>Constants.FV_PGF_CENTER (0x03)</code> <code>Constants.FV_PGF_JUSTIFIED (0x04)</code></li> </ul>
RightIndent	int	The paragraph's right margin, measured from the right side of the current text column.
RightIndentChange	int	Amount by which to increase or decrease the right margin.
SpaceAbove	int	Indicates the space above the paragraph.
SpaceAboveChange	int	Amount by which to increase or decrease the space above.

Property name	Data Type	Description
SpaceBelow	int	Indicates the space below the paragraph.
SpaceBelowChange	int	Amount by which to increase or decrease the space below.
Tabs	<i>Tabs</i>	An array of tab descriptions that specify the positions and types of tab stops in the paragraph.
Capitalization	int	Type of capitalization to use. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAPITAL_CASE_NORM ()</li> <li>Constants.FV_CAPITAL_CASE_SMALL ()</li> <li>Constants.FV_CAPITAL_CASE_LOWER ()</li> <li>Constants.FV_CAPITAL_CASE_UPPER ()</li> </ul>
ChangeBar	int	1 if Change Bars are on;0 otherwise.
Color	<i>Color</i>	The spot color (FO_Color ID).
CombinedFontFamily	<i>CombinedFontDefn</i>	The combined font definition (FO_CombinedFontDefn)
FontAngle	int	Denotes the font angle (specifies an index into the array of font angles provided by the session property FontAngleNames).
FontFamily	int	Denotes the font family (specifies an index into the array of font families provided by the session property FontFamilyNames).



Property name	Data Type	Description
Language	int	<p>The hyphenation and spell-checking language to use. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_LANG_BRAZILIAN (0x0B)</li> <li>Constants.FV_LANG_BRITISH (0x02)</li> <li>Constants.FV_LANG_CANADIAN_FRENCH (0x06)</li> <li>Constants.FV_LANG_CATALAN (0x08)</li> <li>Constants.FV_LANG_DANISH (0x0C)</li> <li>Constants.FV_LANG_DUTCH (0x0D)</li> <li>Constants.FV_LANG_ENGLISH (0x01)</li> <li>Constants.FV_LANG_FINNISH (0x10)</li> <li>Constants.FV_LANG_FRENCH (0x05)</li> <li>Constants.FV_LANG_GERMAN (0x03)</li> <li>Constants.FV_LANG_ITALIAN (0x09)</li> <li>Constants.FV_LANG_LANGUAGE (0x00)</li> <li>Constants.FV_LANG_NORWEGIAN (0x0E)</li> <li>Constants.FV_LANG_NYNORSK (0x0F)</li> <li>Constants.FV_LANG_PORTUGUESE (0x0A)</li> <li>Constants.FV_LANG_SPANISH (0x07)</li> <li>Constants.FV_LANG_SWEDISH (0x11)</li> <li>Constants.FV_LANG_SWISS_GERMAN (0x04)</li> <li>Constants.FV_LANG_JAPANESE (0x12)</li> <li>Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>Constants.FV_LANG_KOREAN</li> </ul>

Property name	Data Type	Description
FontSize	int	The font size (2 pt to 400 pt).
FontSizeChange	int	Amount by which to increase or decrease the font size.
FontVariation	int	Font variation (specifies an index into the array of font variations provided by the session property <code>FontVariationNames</code> ).
FontWeight	int	Font weight (specifies an index into the array of font weights provided by the session property <code>FontWeightNames</code> ).
KernX	int	Horizontal kern value for manual kerning expressed as a percentage of an em (metric – 100% to 1000%). A positive value moves a character right and a negative value moves a character left.
KernY	int	Vertical kern value for manual kerning expressed as a percentage of an em (metric – 100% to 1000%). A positive value moves characters up and a negative value moves characters down.
Overline	int	1 if <code>Overline</code> is enabled. 0, otherwise.
PairKern	int	1 if <code>Pair Kern</code> is enabled. 0, otherwise.
Position	int	Specifies position relative to baseline of text. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_POS_NORM (0): Normal</li> <li>Constants.FV_POS_SUPER (1): Superscript</li> <li>Constants.FV_POS_SUB (2): Subscript</li> </ul>
Stretch	int	Character stretch (set width) expressed as a percentage of normal stretch for the font (metric –10% to 1000%).

Property name	Data Type	Description
StretchChange	int	Amount expressed as a percentage (metric – 10% to 1000%) by which to increase or decrease the character stretch.
Strikethrough	int	1 if <code>Strikethrough</code> is enabled. 0, otherwise.
Underlining	int	Type of underlining. The value is one of: <ul style="list-style-type: none"> <li>• <code>Constants.FV_CB_NO_UNDERLINE (0)</code> <i>Constants.FV_CB_SINGLE_UNDERLINE (1)</i> <i>Constants.FV_CB_DOUBLE_UNDERLINE (2)</i> <i>Constants.FV_CB_NUMERICAL_UNDERLINE (3)</i></li> </ul>
BlockLines	int	The number of Widow/Orphan lines.
KeepWithNext	int	1 if <code>Keep With Next Paragraph</code> is enabled. 0 otherwise.
KeepWithPrev	int	1 if <code>Keep With Previous Paragraph</code> is enabled. 0 otherwise.
Placement	int	Denotes the Paragraph placement. The value is one of: <ul style="list-style-type: none"> <li>• <code>Constants.FV_PGF_SIDE_BODY (0)</code> <i>Constants.FV_PGF_SIDEHEAD_TOP (1)</i> <i>Constants.FV_PGF_SIDEHEAD_FIRST_BASELINE (2)</i> <i>Constants.FV_PGF_SIDEHEAD_LAST_BASELINE (3)</i> <i>Constants.FV_PGF_RUN_IN (4)</i> <i>Constants.FV_PGF_STRADDLE (5)</i> <i>Constants.FV_PGF_STRADDLE_NORMAL_ONLY (6)</i></li> </ul>
RunInSeparator	string	Denotes the string for Run-In Head Default Punctuation.

Property name	Data Type	Description
Start	int	Denotes the Vertical placement of a paragraph. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_ANYWHERE ()</li> <li>Constants.FV_PGF_TOP_OF_COL ()</li> <li>Constants.FV_PG-F_TOP_OF_PAGE ()</li> <li>Constants.FV_PG-F_TOP_OF_LEFT_PAGE ()</li> <li>Constants.FV_PG-F_TOP_OF_RIGHT_PAGE ()</li> </ul>
CellBottomMargin	int	Amount added to default bottom margin of table cell.
CellBottomMarginChange	int	Amount by which to increase or decrease the cell bottom margin.
CellBottomMarginFixed	int	1 if the cell bottom margin is fixed. 0 otherwise.
CellLeftMargin	int	Amount added to default left margin of table cell.
CellLeftMarginChange	int	Amount by which to increase or decrease the cell left margin.
CellLeftMarginFixed	int	1 if the cell left margin is fixed. 0 otherwise.
CellRightMargin	int	Amount added to default right margin of table cell.
CellRightMarginFixed	int	1 if the cell right margin is fixed. 0, otherwise.
CellTopMargin	int	Amount added to default top margin of table cell.
CellTopMarginChange	int	Amount by which to increase or decrease the cell top margin.
CellTopMarginFixed	int	1 if the cell top margin is fixed. 0, otherwise.

Property name	Data Type	Description
CellVAlignment	int	Denotes the vertical alignment of a paragraph when it is the first one in a cell. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_V_ALIGN_TOP (0)</li> <li>Constants.FV_PGF_V_ALIGN_MIDDLE (1)</li> <li>Constants.FV_PGF_V_ALIGN_BOTTOM (2)</li> </ul>
BkColor	Color	Denotes the text background color. If this property is applied on a paragraph format, it denotes the background color of the whole paragraph; if it is applied on a character format, it denotes the background color of the text over which the character format is applied.
PgfBoxColor	Color	Denotes the background color of the paragraph box that encloses the paragraph text.
Direction	int	Set or get the direction of the document. Possible values: <ul style="list-style-type: none"> <li>Constants.FV_DIR_Inherit - Inherit the direction of the parent</li> <li>Constants.FV_DIR_LTR - Left-to-right</li> <li>FV_DIR_RTL - Right-to-left</li> </ul>
ResolvedDirection	int	Get the inherited direction of the document. Possible values: <ul style="list-style-type: none"> <li>Constants.FV_DIR_LTR - Left-to-right</li> <li>FV_DIR_RTL - Right-to-left</li> </ul>

## FmtChangeList methods

*Delete, GetProps, ObjectValid, SetProps.*

## FmtRule

Property name	Data Type	Description
CountElements	<i>Strings</i>	The list of element tags to count among the element's ancestors, if the format rule is a level rule. The tags are specified by the <code>Count ancestors</code> named element of the format rule.
ElementDef	<i>ElementDef</i>	Denotes the ID of the element definition that contains the format rule (FO_ElementDef ID), if the format rule is not nested.
FmtRuleClause	<i>FmtRuleClause</i>	Denotes the ID of the rule clause that contains the format rule (FO_ElementDef ID), if the format rule is nested.
FmtRuleClauses	Objects	IDs of the format rule's format rule clause objects (FO_FmtRuleClause IDs).
FmtRuleType	int	The format rule's type. The value is one of: <ul style="list-style-type: none"> <li>• <code>Constants.FV_CONTEXT_RULE (0)</code></li> <li>• <code>Constants.FV_LEVEL_RULE (1)</code></li> </ul>
StopCountingAt	string	Denotes the the tag of the element at which to stop counting elements, if the the format rule is a level rule. The tag is specified by the <code>Stop counting at first ancestor</code> named element.

### FmtRule methods

*Delete, GetProps, NewFmtRuleClauses, ObjectValid, SetProps.*

## FmtRuleClause

Property name	Data Type	Description
ContextLabel	string	The context label for generated files. The label cannot contain white-space characters or any of these special characters: ( ) &   , * + ? < > % [ ] = ! ; : { } " When a user displays the Set Up dialog box to set up a generated file, the label appears next to elements to which the rule clause applies.
ElemPrefixSuffix	string	The text of the prefix or suffix. ElemPrefixSuffix specifies NULL if there is no prefix or suffix.
FmtChangeList	<i>FmtChangeList</i>	If the format rule clause specifies a format change list (RuleClauseType specifies Constants.FV_RC_CHANGE_LIST ()), FmtChangeList specifies the ID of the format change list (FO_FmtChangeList ID). To change the FmtChangeList property, use the NewFmtRuleObject() property.
FmtChangeListTag	string	Denotes the change list's tag, if the format rule clause specifies a change list (RuleClauseType specifies Constants.FV_RC_CHANGE_LIST_TAG (3)).
FmtRule	<i>FmtRule</i>	The ID of the format rule containing the format rule clause (FO_FmtRule ID).
FormatTag	string	The format tag if the format rule clause specifies one (RuleClauseType specifies Constants.FV_RC_TAG (0)). If IsTextRange is True, FormatTag specifies a character format tag; otherwise it specifies a paragraph tag, table tag, marker type, cross-reference format, or equation size.

Property name	Data Type	Description
IsTextRange	int	1 if the container element is formatted as a text range instead of a paragraph;0 otherwise.
RuleClauseType	int	The type of rule clause. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_RC_TAG (0)</li> <li>Constants.FV_RC_SUB_FMTRUL (1)</li> <li>Constants.FV_RC_CHANGELIST (2)</li> <li>Constants.FV_RC_CHANGE-LIST_TAG (3)</li> </ul>
Specification	string	The format clause's context or level specification.
SpecificationForCSS	<i>Strings</i>	A list of CSS specifications that match the <code>Specification</code> for the current clause. For example, assume an element has a specification of <code>F &lt; (G   H)</code> . This property returns the following strings: <code>G &gt; F &gt; E</code> and <code>H &gt; F &gt; E</code> .
SubFmtRule	<i>FmtRule</i>	Denotes the the format rule's ID (FO_FmtRule ID), if the format rule clause contains a nested format rule (if <code>RuleClauseType</code> specifies <code>Constants.FV_RC_SUB_FMTRULE (1)</code> ),

## FmtRuleClause methods

[Delete](#), [GetProps](#), [NewFmtChangeList](#), [NewSubFmtRule](#), [ObjectValid](#), [SetProps](#).



**Fn**

Property name	Data Type	Description
ContentHeight	int	The distance between the top of the footnote and the baseline of the last line in the footnote.
Element	<i>Element</i>	Specifies the the ID of the element containing the footnote, if the footnote is in a FrameMaker document.
FirstPgf	<i>Pgf</i>	First paragraph in the footnote (FO_Pgf ID).
FnNum	int	Footnote number.
InTextFrame	<i>TextFrame</i>	Text frame containing the footnote (FO_TextFrame ID).
InTextObj	<i>SubCol</i>	Sub column that contains the footnote (FO_SubCol).
LastPgf	<i>Pgf</i>	Last paragraph in the footnote (FO_Pgf ID).
NextFnInDoc	<i>Fn</i>	Next footnote (FO_Fn ID) in the document.
NextFn	<i>Fn</i>	Next footnote in the text frame (FO_Fn ID).
Overflowed	int	1 if the text in the footnote overflows; 0, otherwise.
PrevFn	<i>Fn</i>	Previous footnote in the text frame (FO_Fn ID).
TextLoc	<i>TextLoc</i>	Text location of the footnote symbol.
Unique	int	Footnote's unique ID.
FnAnchorString	string	This is the display string for a footnote anchor (generally a number string). This is a read-only property of a footnote and cannot be set by the user.
UserString	string	A string to which clients can store private data.

## Fn methods

*Delete, GetProps, GetText, ObjectValid, SetProps.*

## Font

Property name	Data Type	Description
family	uint	Index of the font family.
variation	uint	Index of the font variation.
weight	uint	Index of the font weight.
angle	uint	Index of the font angle.

## Font methods

*Font.*

## Fonts

An Array of *Font* objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## Fonts methods

*concat, pop, push.*

## Group

Property name	Data Type	Description
FirstGraphicInGroup	FObject	First object in the group.
LastGraphicInGroup	FObject	Last object in the group.

Property name	Data Type	Description
Angle	int	Angle of the object's rotation.
ArrowBaseAngle	int	Arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Arrowhead tip angle in degrees.
ArrowType	int	Arrowhead style. The value is one of: <ul style="list-style-type: none"> <li>• <code>Constants.FV_ARROW_STICK (0x1)</code> <code>Constants.FV_ARROW_HOLLOW (0x2)</code> <code>Constants.FV_ARROW_FILLED (0x3)</code></li> </ul>
BorderWidth	int	Denotes the border width (0.015 points to 360 points).
Color	<i>Color</i>	The spot color (FO_Color ID).
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a <code>MetricsT</code> structure. The 0th element of the <code>MetricsT.MetricsT_val</code> array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.

Property name	Data Type	Description
Fill	int	The fill pattern (numbers between 0 and 15). The ESTK provides constants for the following fill patterns: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR(15)</li> </ul>
FrameParent	FMOBJECT	Frame containing the graphic object (FO_AFrame or FO_UnanchoredFrame ID).
GraphicCantBeSelected	int	1 if the graphic object cannot be selected; 0, otherwise.
GraphicsSelected	int	1 if the graphic object is selected; 0, otherwise.
GroupParent	Group	Group that the object is in (FO_Group ID). <i>NOTE: Anchored and unanchored frames do not have this property.</i>
HeadArrow	int	1 if the line has an arrowhead; 0, otherwise.
Height	int	Height of the object (0.125 points to 3600 points).
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Type of line end. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>

Property name	Data Type	Description
LocX	int	Distance from the left side of the parent frame (–216 inches to 216 inches).If the graphic object is an anchored frame, the distance is calculated from the left side of the page frame. You cannot set LocX for anchored frames.
LocY	int	Distance from the top of the parent frame (–216 inches to 216 inches).If the graphic object is an anchored frame, the distance is calculated from the top of the page frame. You cannot set LocY for anchored frames.
NextGraphicInDoc	FMOBJECT	Next graphic object in the document.
NextGraphicInFrame	FMOBJECT	Next graphic object in the frame.
NextGraphicInGroup	FMOBJECT	Next graphic object in the group.
NextSelectedGraphicInDoc	FMOBJECT	Next selected graphic object in document.
ObjectAttributes	<i>Strings</i>	A list of strings, each string expressing an attribute that is specified for an anchored frame in the <code>Object Properties &gt; Object Attributes</code> dialog box.Each string is in the form of <code>&lt;tag&gt;string_text&lt;/tag&gt;</code> . You can use the backslash to escape characters such as <code>\n</code> , <code>\r</code> , or <code>\t</code> for newline, carriage return, and tab, respectively. For a backslash character in the string text, type <code>\\</code> .
Overprint	int	Specifies the overprint settings for the object. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>

Property name	Data Type	Description
Pen	int	The pen pattern (numbers between 0 and 7). The ESTK provides constants for the following values: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FMOBJECT	Previous graphic object in the frame.
PrevGraphicInGroup	FMOBJECT	Previous graphic object in the group
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the the width of the runaround gapfor a runaround object.
TailArrow	int	1 if arrowhead is present at the beginning of the line. 0, otherwise.
TintPercent	int	The tint percentage.
Unique	int	Unique ID of the object.
Width	int	Width of the object (0.125 points to 3600 points).

## Group methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## HiddenPage

Property name	Data Type	Description
Name	string	Name of the hidden page.

Property name	Data Type	Description
PageFrame	<i>UnanchoredFrame</i>	Denotes the page frame (FO_UnanchoredFrame ID)
PageHeight	int	Height of the hidden page.
PageWidth	int	Width of the hidden page.

## HiddenPage methods

*ApplyPageLayout, GetProps, ObjectValid, SetProps.*

## GraphicsFmt

Property name	Data Type	Description
UseColumnsAreBalanced	int	1 if terminal and underfilled columns in the flow are balanced.
UseSideHeadPlacement	int	Placement of side heads relative to columns in the text frame: FV_SH_LEFT FV_SH_RIGHT FV_SH_INSIDE FV_SH_OUTSIDE
UseRadius	MetricT	Radius of corner; 0 for a square corner
UseFlowsAutoConnect	Int	1 if Autoconnect is enabled.
UseFlowsPostScript	int	1 if flow is PostScript code.
UseBorderWidth	MetricT	Border width (0.015 pt to 360 pt).
UseFill	int	The fill pattern (numbers between 0 and 15). The FDK provides constants for several fill patterns: FV_FILL_BLACK FV_FILL_WHITE FV_FILL_CLEAR

Property name	Data Type	Description
UsePen	int	The pen pattern (numbers between 0 and 15). The FDK provides constants for several pen patterns: FV_FILL_BLACK FV_FILL_WHITE FV_FILL_CLEAR
UseRunaroundGap	MetricT	If the object is a runaround object, the width of the runaround gap.
UseTintPercent	MetricT	The tint percentage
UseOverprint	int	Specifies the overprint settings for the object: FV_OVERPRINT FV_KNOCKOUT FV_FROMCOLOR
UseAngle	MetricT	Angle of the object's rotation.
UseLocX	MetricT	Distance from the left side of the parent frame.
UseLocY	MetricT	Distance from the top side of the parent frame
UseWidth	MetricT	Width of object
UseHeight	MetricT	Height of object.
UseRunaround	MetricT	If the object is a runaround object, the width of the runaround gap.

## InlineComponent

A component that can be placed inline within a Frammaker document. Currently FrameMaker supports the mini-TOC inline component.

Property name	Data Type	Description
InlineComponentType	int	Denotes the type of inline component. (FV_MiniTOC).
ExtractTags	Strings	Paragraph tags to include for the inline component



Property name	Data Type	Description
InsertLinks	int	Flag that specifies if hyperlinks need to be created in the inline component.
TextRange	<i>TextRange</i>	The text range that the inline component encompasses.
Unique	int	The inline component's unique ID
NextInlineComponentInDoc	InlineComponent	The next instance of an inline component in the document. Currently only the mini-TOC inline component can exist in a document.

### InlineComponent methods

*Delete, GetProps, ObjectValid, SetProps.*

### Inset methods

*Delete, DeletePropByName, GetIntByName, GetMetricByName, GetProps, GetUBytesByName, ObjectValid, SetIntByName, SetMetricByName, SetProps, SetUBytesByName.*

### IdValuePair

Property name	Data Type	Description
ID	int	Specifies the unique identifier for the value pair.
value	string or int	Specifies the value for the value pair.

### IdValuePair methods

### IdValuePairs

An Array of objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## Inset

Property name	Data Type	Description
InsetDpi	int	Specifies the scaling information for bitmap file (corresponds to the value specified in the Image File Scaling Options dialog box when the graphics file is imported).
ImportHint	string	Record identifying the filter used to import the graphic. FrameMaker uses this record to find the filter to use when updating the inset. <i>NOTE: The <code>ImportHint</code> property does not apply to graphics imported by copy. FrameMaker uses the facet name stored with the graphic to identify the filter that filtered a graphic imported by copy.</i>
InsetEditor	string	Name of the application to call to edit inset or imported object.
InsetGfxActiveInPdf	BoolString	If this property is set, on publishing a document to PDF, the inset object that has facets FLV, U3D, or SWF will be activated as soon as the page containing the graphic object is visible. In PDF the graphic objects are called annotation.
InsetGfxName	String	Assigns a name to a graphic object. It will work only in case of inset objects that have an FLV, U3D or SWF facet. The name of the graphic should not contain any special characters or spaces.

Property name	Data Type	Description
InsetGfxPlayWindowInPdf	Bool	If this property is set, on publishing a document to PDF, the inset object that has facets FLV, U3D, or SWF will be activated in a new window in a PDF file. In PDF, the graphic objects are called annotation.
InsetFile	string	Platform-specific pathname if the inset is an external inset, or a null string (" ") if it is internal. The pathname can be document-relative.
InsetIsFixedSize	int	1 if scaling of bitmap file is inhibited. 0, otherwise.
InsetIsFlippedSideways	int	1 if inset is flipped about the vertical axis. 0, otherwise.
InsetJavaScriptAttached	Bool	Denotes whether or not Javascript is attached with the graphic object that has a U3D facet.
InsetJavaScriptFile	String	Attaches the given Javascript file to the graphic object that has a U3D facet. If the value of the file path is null, the Javascript attached to the inset is removed.
InsetMonikerFilePath	String	Provides the file path of the moniker of an inset object that has an OLE2 facet.
InsetSaveFacetToFile	Strings	Saves the given facet of an inset to a given file. This is set-only property and no get operations are possible. The input to the method contain the property name and list of strings. In the list of strings the, first strings is the facet name and the second string is the filename where the facet is to be saved.
InsetU3dAnimationList	Strings	Provides the list of "animations" defined in the U3D facet of an inset object.
InsetU3dPartList	Strings	Provides the list of "parts" defined in the U3D facet of an inset object.

Property name	Data Type	Description
InsetU3dViewList	Strings	Retrieves the list of "views" defined in the U3D facet of an inset object.
InsetUpdater	string	Not currently implemented.
PageNum	int	The page number of the PDF file to display in the inset, when importing a PDF file. <i>NOTE: PDF page numbering begins with 0.</i>
InsetPosterFile	string	Sets the specified file as a poster on the Inset. This property is not supported for all the inset objects but supported only for multimedia objects and u3d object. The property takes the poster file path as the value.
InsetGfxActiveInPdf	string	If this property is set, on publishing a document to PDF, the inset object (which has facets FLV, U3d, or swf) will be activated as soon as the page containing the graphic object is visible. In PDF the graphic objects are called annotation.
InsetJavaScriptFile	string	Attaches the given Javascript file to the graphic object that has a U3d facet. If the value of the file path is null, the Javascript attached to the inset is removed.
InsetU3dAnimationList	string	A read-only property. Provides the list of "animations" defined on the U3d facet of a inset object.
InsetU3dPartList	string	Provides the list of "parts" defined in the U3d facet of a inset object.
InsetU3dViewList	string	A read-only property. Retrieves the list of "views" defined in the U3d facet of a inset object.
Angle	int	Angle of the object's rotation.
ArrowBaseAngle	int	Arrowhead base angle in degrees.

Property name	Data Type	Description
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Arrowhead tip angle in degrees.
ArrowType	int	Arrowhead style. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Denotes the border width (0.015 points to 360 points).
Color	<i>Color</i>	The spot color (FO_Color ID)
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a <code>MetricsT</code> structure. The 0th element of the <code>MetricsT.MetricsT_val</code> array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements
Fill	int	The fill pattern (numbers between 0 and 15). The ESTK provides constants for the following fill patterns: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR(15)</li> </ul>

Property name	Data Type	Description
FrameParent	FObject	Frame containing the graphic object (FO_AFrame or FO_UnanchoredFrame ID).
GraphicCantBeSelected	int	1 if the graphic object cannot be selected; 0, otherwise.
GraphicsSelected	int	1 if the graphic object is selected; 0, otherwise.
GroupParent	<i>Group</i>	Group that the object is in (FO_Group ID). <b>NOTE:</b> <i>Anchored and unanchored frames do not have this property.</i>
HeadArrow	int	1 if the line has an arrowhead; 0, otherwise.
Height	int	Height of the object (0.125 points to 3600 points).
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Type of line end. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Distance from the left side of the parent frame (-216 inches to 216 inches). If the graphic object is an anchored frame, the distance is calculated from the left side of the page frame. You cannot set LocX for anchored frames.

Property name	Data Type	Description
LocY	int	Distance from the top of the parent frame (–216 inches to 216 inches). If the graphic object is an anchored frame, the distance is calculated from the top of the page frame. You cannot set <code>LocY</code> for anchored frames.
NextGraphicInDoc	FObject	Next graphic object in the document.
NextGraphicInFrame	FObject	Next graphic object in the frame.
NextGraphicInGroup	FObject	Next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Next selected graphic object in document.
ObjectAttributes	<i>Strings</i>	A list of strings, each string expressing an attribute that is specified for an anchored frame in the <code>Object Properties &gt; Object Attributes</code> dialog box. Each string is in the form of <code>&lt;tag&gt;string_text&lt;/tag&gt;</code> . You can use the backslash to escape characters such as <code>\n</code> , <code>\r</code> , or <code>\t</code> for newline, carriage return, and tab, respectively. For a backslash character in the string text, type <code>\\</code> .
Overprint	int	Specifies the overprint settings for the object. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	The pen pattern (numbers between 0 and 7). The ESTK provides constants for the following values: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Previous graphic object in the frame.
PrevGraphicInGroup	FObject	Previous graphic object in the group

Property name	Data Type	Description
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the the width of the runaround gapfor a runaround object.
TailArrow	int	1 if arrowhead is present at the beginning of the line. 0, otherwise.
TintPercent	int	The tint percentage.
Unique	int	Unique ID of the object.
Width	int	Width of the object (0.125 points to 3600 points).

## Inset methods

[Delete](#), [DeletePropByName](#), [GetIntByName](#), [GetMetricByName](#), [GetProps](#), [GetUBytesByName](#), [ObjectValid](#), [SetIntByName](#), [SetMetricByName](#), [SetProps](#), [SetUBytesByName](#).

## Ints

An Array of objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## Ints methods

[concat](#), [pop](#), [push](#).



## KeyCatalog

Property name	Data Type	Description
Name	string	Name of the KeyCatalog
IsDefault	bool	If True, the Key Catalog is the default one for the current workflow. If False, Key Catalog is not the default one for the current workflow.
IsStale	bool	If True, the Key Catalog is made as stale and needs to be re-loaded before using. If False, the Key Catalog is not stale and can be used.
NextKeyCatalogInSession	F_ObjHandle T	Next Key Catalog in the session.
NotLoaded	bool	If True, the Key Catalog is not loaded and cannot be used. If False, the Key Catalog is loaded and can be used.
Source	string	Complete path of the file conatining the Key Catalog.
SourceType	FV_KeySrcType peNone	FV_KeySrcTypeDitamap Type of the file conatining the Key Catalog.
KeyCount	int	Number of keys in the Key Catalog including duplicate definitons.
KeyCatalogClientName	string	Name of the client owning the key catalog.

## Line

Property name	Data Type	Description
NumPoints	int	Specifies the number of vertices. The default is 2 (the line's start point and end point).

Property name	Data Type	Description
Points	<i>Points</i>	Array of x-y coordinate pairs that specify the line's vertices. The default coordinate pairs are for the line's start point and end point.
Angle	int	Angle of the line's rotation.
ArrowBaseAngle	int	Arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Arrowhead tip angle in degrees.
ArrowType	int	Arrowhead style. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Denotes the border width (0.015 points to 360 points).
Color	<i>Color</i>	The spot color (FO_Color ID)
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a <code>MetricsT</code> structure. The 0th element of the <code>MetricsT.MetricsT_val</code> array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements

Property name	Data Type	Description
Fill	int	The fill pattern (numbers between 0 and 15). The ESTK provides constants for the following fill patterns: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR(15)</li> </ul>
FrameParent	FMObject	Frame containing the graphic object (FO_AFrame or FO_UnanchoredFrame ID).
GraphicCantBeSelected	int	1 if the graphic object cannot be selected; 0, otherwise.
GraphicsSelected	int	1 if the graphic object is selected; 0, otherwise.
GroupParent	<i>Group</i>	Group that the line is in (FO_Group ID).
HeadArrow	int	1 if the line has an arrowhead; 0, otherwise.
Height	int	Height of the object (0.125 points to 3600 points).
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Type of line end. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>

Property name	Data Type	Description
LocX	int	Distance from the left side of the parent frame (–216 inches to 216 inches).If the graphic object is an anchored frame, the distance is calculated from the left side of the page frame. You cannot set LocX for anchored frames.
LocY	int	Distance from the top of the parent frame (–216 inches to 216 inches).If the graphic object is an anchored frame, the distance is calculated from the top of the page frame. You cannot set LocY for anchored frames.
NextGraphicInDoc	FMOBJECT	Next graphic object in the document.
NextGraphicInFrame	FMOBJECT	Next graphic object in the frame.
NextGraphicInGroup	FMOBJECT	Next graphic object in the group.
NextSelectedGraphicInDoc	FMOBJECT	Next selected graphic object in document.
ObjectAttributes	<i>Strings</i>	A list of strings, each string expressing an attribute that is specified for an anchored frame in the <code>Object Properties &gt; Object Attributes</code> dialog box.Each string is in the form of <code>&lt;tag&gt;string_text&lt;/tag&gt;</code> . You can use the backslash to escape characters such as <code>\n</code> , <code>\r</code> , or <code>\t</code> for newline, carriage return, and tab, respectively. For a backslash character in the string text, type <code>\\</code> .
Overprint	int	Specifies the overprint settings for the object. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>

Property name	Data Type	Description
Pen	int	The pen pattern (numbers between 0 and 7). The ESTK provides constants for the following values: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FMOBJECT	Previous graphic object in the frame.
PrevGraphicInGroup	FMOBJECT	Previous graphic object in the group
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the the width of the runaround gapfor a runaround object.
TailArrow	int	1 if arrowhead is present at the beginning of the line. 0, otherwise.
TintPercent	int	The tint percentage.
Unique	int	Unique ID of the object.
Width	int	Width of the object (0.125 points to 3600 points).

## Line methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## Marker

Property name	Data Type	Description
Element	<i>Element</i>	Denotes the the element containing the marker, if the marker is a structured marker in a FrameMaker document,.
MarkerText	string	The marker's text string.
MarkerTypeid	<i>MarkerType</i>	The ID of the current marker's type (FO_MarkerType).
NextMarkerInDoc	<i>Marker</i>	Denotes the next marker (FO_Marker ID).
OldTypeNum	int	A number to map markers from documents earlier than version 5.5 to this marker type.
TextLoc	<i>TextLoc</i>	Text location of the marker's symbol.
Unique	int	The marker's unique ID.
UserString	string	A string to which clients can store private data.

### Marker methods

*Delete, GetProps, ObjectValid, SetProps.*

## MarkerType

Property name	Data Type	Description
NextMarkerTypeInDoc	<i>MarkerType</i>	Denotes the next marker type (FO_MarkerType ID).
Name	string	The name of this marker type, as it appears in the user interface.

Property name	Data Type	Description
InvariantName	string	An internal name for the marker type. By default, this is the same as Name. However, this can differ from Name if the user interface is in another language.
Public	bool	True if the marker type should appear in the user interface. The default is True.
Transient	bool	True if markers of this type should not be saved to files. The default is False.
Required	bool	True if the marker type is required by FrameMaker. The default is False.

## MarkerType methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## MasterPage

Property name	Data Type	Description
Name	string	Name of master page (for example, Right or Left).
PageFrame	<a href="#">UnanchoredFrame</a>	Page frame (FO_UnanchoredFrame ID).
PageHeight	int	Height of the page.
PageNext	<a href="#">MasterPage</a>	Next master page (FO_MasterPage ID) in the document.
PageNum	int	Page number.
PagePrev	<a href="#">MasterPage</a>	Previous master page (FO_MasterPage ID) in the document.
PageWidth	int	Width of the page.

## MasterPage methods

[ApplyPageLayout](#), [Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

# Math

Property name	Data Type	Description
BasePointX	int	Horizontal placement of text line base point relative to the left side of the frame.
BasePointY	int	Vertical placement of text line base point relative to top of the frame.
MathFullForm	string	String representing the mathematical expression.
MathSize	int	The equation size. The value is one of : <ul style="list-style-type: none"> <li>Constants.FV_MATH_MEDIUM (0)</li> <li>Constants.FV_MATH_SMALL (1)</li> <li>Constants.FV_MATH_LARGE (2)</li> </ul>
TextLineType	int	Type of text line. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TEXTLINE_LEFT (0)</li> <li>Constants.FV_TEXTLINE_RIGHT (1)</li> <li>Constants.FV_TEXTLINE_CENTER (2)</li> <li>Constants.FV_TEXTLINE_MATH (3)</li> </ul>
Angle	int	Specifies the angle of the object's rotation.
ArrowBaseAngle	int	Indicates the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleHead	int	Set to 1 if the arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to <code>False</code> .
ArrowTipAngle	int	Specifies the arrowhead tip angle in degrees.



Property name	Data Type	Description
ArrowType	int	Denotes the Arrowhead style. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x01)</li> <li>Constants.FV_ARROW_HOLLOW (0x02)</li> <li>Constants.FV_ARROW_FILLED (0x03)</li> </ul>
BorderWidth	int	Specifies the border width in points. The permissible range is 0.015 point to 360 point.
Color	<i>Color</i>	Indicates the spot color.
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a Metrics object. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern. The patterns are denoted by numbers between 0 and 15. The following constants are provided for the fill patterns 0, 7 and 15: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Specifies the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Else, set to 0.
GraphicsSelected	int	Set to 1 if the graphic object is selected, else set to 0.

Property name	Data Type	Description
GroupParent	<i>Group</i>	Indicates the group to which the object belongs. Anchored and unanchored frames do not have this property.
HeadArrow	int	Set to 1 if line has an arrowhead, set to 0 otherwise.
Height	int	Specifies the height of the object in points. The permissible range is from 0.125 pt to 3600 pt.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Indicates the type of line end. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent, in inches. The permissible range is from -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set the LocX property for anchored frames.

Property name	Data Type	Description
LocY	int	Specifies the distance of the object from the top of the parent frame, in inches. The permissible range is from –216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You CANNOT set the LocY property for anchored frames.
NextGraphicInDoc	FObject	Indicates the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Indicates the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Specifies the next selected graphic object in the document.
ObjectAttributes	Strings	Contains a list of strings, each string expressing an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>

Property name	Data Type	Description
Pen	int	Indicates the pen pattern, denoted by numbers between 0 and 7. Constants are provided for some of the fill patterns: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FMOBJECT	Specifies the previous graphic object in the frame.
PrevGraphicInGroup	FMOBJECT	Indicate the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line. Set to 0, otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Denotes the internal identifier of the graphic object.
Width	int	Specifies the width of the object in points. The permissible range is 0.125 point to 3600 point.

## Math methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

# MathML

Property name	Data Type	Description
MathMLDpi	int	Specifies the display Dpi for the MathML object.
MathMLIsFixedSize	int	1 if scaling of bitmap file is inhibited. 0, otherwise.
MathMLIsInverted	int	1 if image is flipped about the horizontal axis. 0, otherwise.
MathMLIsFlippedSideways	int	1 if image is flipped about the vertical axis. 0, otherwise.
MathMLData	strings	Specifies the XML data of the MathML object as a string array.
MathMLComposeDpi	int	Specifies the compose Dpi for the MathML object
MathMLFontSize	int	Specifies the font size for the MathML object
MathMLXmlData	string	Specifies the XML data of the MathML object as a string.
Angle	int	Specifies the angle of the object's rotation.
ArrowBaseAngle	int	Indicates the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleHead	int	Set to 1 if the arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to <code>False</code> .
ArrowTipAngle	int	Specifies the arrowhead tip angle in degrees.

Property name	Data Type	Description
ArrowType	int	Denotes the Arrowhead style. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x01)</li> <li>Constants.FV_ARROW_HOLLOW (0x02)</li> <li>Constants.FV_ARROW_FILLED (0x03)</li> </ul>
BorderWidth	int	Specifies the border width in points. The permissible range is 0.015 point to 360 point.
Color	<i>Color</i>	Indicates the spot color.
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a Metrics object. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern. The patterns are denoted by numbers between 0 and 15. The following constants are provided for the fill patterns 0, 7 and 15: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Specifies the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Else, set to 0.
GraphicsSelected	int	Set to 1 if the graphic object is selected, else set to 0.

Property name	Data Type	Description
GroupParent	<i>Group</i>	Indicates the group to which the object belongs. Anchored and unanchored frames do not have this property.
HeadArrow	int	Set to 1 if line has an arrowhead, set to 0 otherwise.
Height	int	Specifies the height of the object in points. The permissible range is from 0.125 pt to 3600 pt.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Indicates the type of line end. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent, in inches. The permissible range is from -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set the LocX property for anchored frames.

Property name	Data Type	Description
LocY	int	Specifies the distance of the object from the top of the parent frame, in inches. The permissible range is from –216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You CANNOT set the LocY property for anchored frames.
NextGraphicInDoc	FObject	Indicates the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Indicates the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Specifies the next selected graphic object in the document.
ObjectAttributes	<i>Strings</i>	Contains a list of strings, each string expressing an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>



Property name	Data Type	Description
Pen	int	Indicates the pen pattern, denoted by numbers between 0 and 7. Constants are provided for some of the fill patterns: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Specifies the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Indicate the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line. Set to 0, otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Denotes the internal identifier of the graphic object.
Width	int	Specifies the width of the object in points. The permissible range is 0.125 point to 3600 point.

## MathML methods

[Delete](#), [GetProps](#), [SetProps](#), [ObjectValid](#).

# Menu

Property name	Data Type	Description
FirstMenuItemInMenu	FObject	The first menu item in the menu.
MenuType	int	Type of menu. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_MENU_MENUBAR (1) - A menu bar defined by the FrameMaker product.</li> <li>Constants.FV_MENU_POPUP (2) - A pop-up menu.</li> <li>Constants.FV_MENU_DEFAULT (4) - A pull-down or pull-right menu.</li> </ul>
Label	string	The label the user sees on a menu. The label for menu item separators is read-only; it is always ---.
MenuItemIsEnabled	int	1 if the menu item is enabled. 0, if it is disabled.
Name	string	The command, menu, or menu item separator name. The names for the default, predefined separators are !Separator, !Separator1, !Separator2, !Separator3, !Separator4, and !Separator5.
NextMenuItemInMenu	FObject	The next menu item, menu, or separator in the menu.
NextMenuItemInSession	FObject	The next menu item, menu, or separator in the list of menu items, menus, and separators in the session.
PrevMenuItemInMenu	FObject	The previous menu item, menu, or separator in the list of menu items, menus, and separators in the session.

## Menu methods

[AddCommandToMenu](#), [AddMenuToMenu](#), [DefineAndAddCommand](#), [DefineAndAddMenu](#), [Delete](#), [Delete](#), [GetProps](#), [MenuItemInMenu](#), [ObjectValid](#), [SetProps](#).

## MenuItemSeparator

Property name	Data Type	Description
Label	string	The label for menu item separators is read-only; it is always ---.
MenuItemEnabled	int	1 if the menu item separator is enabled. 0, if it is disabled.
Name	string	The menu item separator name. The names for the default, predefined separators are !Separator, !Separator1, !Separator2, !Separator3, !Separator4, and !Separator5.
NextMenuItemInMenu	FObject	The next menu item separator in the menu.
NextMenuItemInSession	FObject	The next menu item separator in the list of menu items, menus, and separators in the session.
PrevMenuItemInMenu	FObject	The previous menu item separator in the list of menu items, menus, and separators in the session.

### MenuItemSeparator methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## Metrics

An Array of objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

### Metrics methods

[concat](#), [pop](#), [push](#).

# Pgf

Property name	Data Type	Description
MinJRomSpace	int	Minimum Asian-Roman space.
OptJRomSpace	int	Optimum Asian-Roman space.
MaxJRomSpace	int	Maximum Asian-Roman space.
MinJLetSpace	int	Minimum Asian letter space.
OptJLetSpace	int	Optimum Asian letter space.
MaxJLetSpace	int	Maximum Asian letter space.
YakumonoType	int	The Yakumono rules to handle punctuation characters. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_FLOATING_YAKUMONO (0x00)</li> <li>Constants.FV_MONOSPACE_YAKUMONO (0x01)</li> <li>Constants.FV_FIXED_YAKUMONO (0x02)</li> </ul>
AutoNumChar	string	Character format for the automatic numbering string specified by <code>AutoNumString</code> ; "" if the default character format is used
AutoNumString	string	Autonumber format string (for example, <n>.<n+>)
NumAtEnd	int	1 if numbering position is End of Paragraph; 0 if it is Beginning of Paragraph.
PgflsAutoNum	int	1 if autonumbering is enabled.
PgfNumber	string	The formatted string representation of the paragraph number; for example, 1.2 for a paragraph whose <code>AutoNumString</code> property is set to <n>.<n+>

Property name	Data Type	Description
Capitalization	int	Type of capitalization to use. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAPITAL_CASE_NORM (0)</li> <li>Constants.FV_CAPITAL_CASE_SMALL (1)</li> <li>Constants.FV_CAPITAL_CASE_LOWER (2)</li> <li>Constants.FV_CAPITAL_CASE_UPPER (3)</li> </ul>
ChangeBar	int	1 if Change Bars are on.
Color	<i>Color</i>	Spot color (FO_Color ID).
CombinedFontFamily	<i>CombinedFontDefn</i>	Combined font definition (FO_CombinedFontDefn).
FontAngle	int	Font angle (specifies an index into the array of font angles provided by the session property <code>FontAngleNames</code> ).
FontEncodingName	string	The font's encoding.
FontFamily	int	Font family (specifies an index into the array of font families provided by the session property <code>FontFamilyNames</code> ).
FontPlatformName	string	Name that uniquely identifies a font on a specific platform.
FontPostScriptName	string	Name given to a font when it is sent to a PostScript printer.
FontSize	int	Font size (2 pt to 400 pt).
FontVariation	int	Font variation (specifies an index into the array of font variations provided by the session property <code>FontVariationNames</code> ).
FontWeight	int	Font weight (specifies an index into the array of font weights provided by the session property <code>FontWeightNames</code> ).

Property name	Data Type	Description
KernX	int	Horizontal kern value for manual kerning expressed as a percentage of an em (metric –100% to 1000%). A positive value moves a character right and a negative value moves a character left.
KernY	int	Vertical kern value for manual kerning expressed as a percentage of an em (metric –100% to 1000%). A positive value moves characters up and a negative value moves characters down.
Overline	int	1 if Overline is enabled.
PairKern	int	1 if Pair Kern is enabled.
Position	int	Specifies position relative to baseline of text. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_POS_NORM (0): Normal</li> <li>Constants.FV_POS_SUPER (1): Superscript</li> <li>Constants.FV_POS_SUB (2): Subscript</li> </ul>
Stretch	int	Character stretch (set width) expressed as a percentage of normal stretch for the font (metric –10% to 1000%).
Strikethrough	int	1 if Strikethrough is enabled.
Underlining	int	Type of underlining. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CB_NO_UNDERLINE (0)</li> <li>Constants.FV_CB_SINGLE_UNDERLINE (1)</li> <li>Constants.FV_CB_DOUBLE_UNDERLINE (2)</li> <li>Constants.FV_CB_NUMERIC_UNDERLINE (3)</li> </ul>
CellBottomMargin	int	Amount added to default bottom margin of table cell.
CellLeftMargin	int	Amount added to default left margin of table cell.

Property name	Data Type	Description
CellMarginsFixed	int	<p>Specifies which cell margins are fixed. The following values can be ORed into it:</p> <ul style="list-style-type: none"> <li>• Constants.FV_PGF_FIXED_L_MARGIN (0x01): the left margin is fixed.</li> </ul> <p><i>Constants.FV_PGF_FIXED_B_MARGIN (0x02): the bottom margin is fixed.</i></p> <p><i>Constants.FV_PGF_FIXED_R_MARGIN (0x04): the right margin is fixed.</i></p> <p><i>Constants.FV_PGF_FIXED_T_MARGIN (0x08): the top margin is fixed.</i></p> <p>If the margin for a cell is fixed, the margin property specifies the absolute value of the cell margin. For example, if Constants.FV_PGF_FIXED_B_MARGIN (0x02) is set, CellBottomMargin specifies the absolute value of the cell's bottom margin, overriding the cell margin specified by the table format.</p> <p>If Constants.FV_PGF_FIXED_B_MARGIN (0x02) is not set, CellBottomMargin is added to the margin specified by the table format.</p>
CellRightMargin	int	Amount added to default right margin of table cell.
CellTopMargin	int	Amount added to default top margin of table cell.

Property name	Data Type	Description
CellVAlignment	int	Vertical alignment of a paragraph when it is the first one in a cell. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_V_ALIGN_TOP (0)</li> <li>Constants.FV_PG-F_V_ALIGN_MIDDLE (1)</li> <li>Constants.FV_PG-F_V_ALIGN_BOTTOM (2)</li> </ul>
Locked	int	1 if the paragraph is part of a text inset that retains formatting information from the source document. The paragraph is not affected by global formatting performed on the document.
FormatOverride	int	1 if the paragraph contains a paragraph format override.
AdjHyphens	int	Number of allowable adjacent hyphens.
Hyphenate	int	1 if Automatic Hyphenation is enabled.
HyphMinPrefix	int	Minimum number of letters that must precede hyphen.
HyphMinSuffix	int	Minimum number of letters that must follow a hyphen.
HyphMinWord	int	Minimum length of a hyphenated word.



Property name	Data Type	Description
Language	int	<p>Hyphenation and spell-checking language to use. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li>Constants.FV_LANG_ENGLISH (0x01)</li> <li>Constants.FV_LANG_BRITISH (0x02)</li> <li>Constants.FV_LANG_GERMAN (0x03)</li> <li>Constants.FV_LANG_SWISS_GERMAN (0x04)</li> <li>Constants.FV_LANG_FRENCH (0x05)</li> <li>Constants.FV_LANG_CANADIAN_FRENCH (0x06)</li> <li>Constants.FV_LANG_SPANISH (0x07)</li> <li>Constants.FV_LANG_CATALAN (0x08)</li> <li>Constants.FV_LANG_ITALIAN (0x09)</li> <li>Constants.FV_LANG_FINNISH (0x10)</li> <li>Constants.FV_LANG_SWEDISH (0x11)</li> <li>Constants.FV_LANG_JAPANESE (0x12)</li> <li>Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>Constants.FV_LANG_KOREAN (0x15)</li> <li>Constants.FV_LANG_PORTUGUESE (0x0A)</li> <li>Constants.FV_LANG_BRAZILIAN (0x0B)</li> <li>Constants.FV_LANG_DANISH (0x0C)</li> <li>Constants.FV_LANG_DUTCH (0x0D)</li> </ul>
		<p>Constants.FV_LANG_NORWEGIAN (0x0E)</p> <p>Constants.FV_LANG_NYNORSK</p>

Property name	Data Type	Description
PgfSpellChecked	int	1 if paragraph has been spell-checked
Unique	int	The paragraph's user ID.
FirstIndent	int	First-line left margin, measured from left side of current text column (0 cm to 100 cm).
LeftIndent	int	Left margin, measured from left side of current text column (0 cm to 100 cm).
RightIndent	int	Right margin, measured from right side of current text column.
LastLineRightIndent	int	Right margin of the last line in the paragraph, measured from right side of the current text column.
Leading	int	Space below each line in a paragraph.
LineSpacing	int	Space between lines in a paragraph measured from baseline to baseline. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_FIXED (0x00): default font size</li> <li>Constants.FV_PGF_PROPORTIONAL (0x01): largest font in line</li> <li>Constants.FV_PGF_FLOATING (0x02): largest ascender in line</li> </ul>
BlockLines	int	The number of Widow/Orphan lines.
KeepWithNext	int	1 if Keep With Next Paragraph is enabled.
KeepWithPrev	int	1 if Keep With Previous Paragraph is enabled.
PgfAlignment	int	Horizontal alignment of paragraph. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_LEFT (0x01)</li> <li>Constants.FV_PGF_RIGHT (0x02)</li> <li>Constants.FV_PGF_CENTER (0x03)</li> <li>Constants.FV_PGF_JUSTIFIED (0x04)</li> </ul>

Property name	Data Type	Description
Placement	int	Paragraph placement. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_SIDEBODY (0)</li> <li>Constants.FV_PGF_SIDEHEAD_TOP (1)</li> <li>Constants.FV_PGF_SIDEHEAD_FIRST_BASELINE (2)</li> <li>Constants.FV_PGF_SIDEHEAD_LAST_BASELINE (3)</li> <li>Constants.FV_PGF_RUN_IN (4)</li> <li>Constants.FV_PGF_STRADDLE (5)</li> <li>Constants.FV_PGF_STRADDLE_NORMAL_ONLY (6)</li> </ul>
RunInSeparator	string	String for Run-In Head Default Punctuation.
SpaceAbove	int	Space above paragraph.
SpaceBelow	int	Space below paragraph.
Start	int	Vertical placement of paragraph. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_ANYWHERE (0x00)</li> <li>Constants.FV_PGF_TOP_OF_COL (0x01)</li> <li>Constants.FV_PGF_TOP_OF_PAGE (0x02)</li> <li>Constants.FV_PGF_TOP_OF_LEFT_PAGE (0x03)</li> <li>Constants.FV_PGF_TOP_OF_RIGHT_PAGE (0x04)</li> </ul>
InTextFrame	<i>TextFrame</i>	Text frame containing the paragraph (FO_TextFrame ID).
InTextObj	FMOBJECT	Subcolumn, footnote, or table cell the paragraph begins in (FO_SubCol, FO_Fn, or FO_Cell ID).
NextPgfInDoc	<i>Pgf</i>	Next paragraph in the document (FO_Pgf ID).

Property name	Data Type	Description
NextPgflnFlow	<i>Pgf</i>	Next paragraph in the flow (FO_Pgf ID).
PrevPgflnFlow	<i>Pgf</i>	Previous paragraph in the flow (FO_Pgf ID).
BottomSeparator	string	Name of frame to put below paragraph.
TopSeparator	string	Name of frame to put above paragraph.
NumTabs	int	Number of tabs in the paragraph..
Tabs	<i>Tabs</i>	Array of tab descriptions that specify the positions and types of tab stops.
Name	string	Name of paragraph format.
NextTag	string	Tag for new next paragraph.
UseNextTag	int	1 if Next Paragraph Tag is enabled.
LetterSpace	int	1 if Word Spacing is enabled.
MaxSpace	int	Maximum word spacing (percentage of an em space in current font).
MinSpace	int	Minimum word spacing (percentage of an em space in current font).
OptSpace	int	Optimum word spacing.
BkColor	<i>Color</i>	Denotes the text background color. If this property is applied on a paragraph format, it denotes the background color of the whole paragraph; if it is applied on a character format, it denotes the background color of the text over which the character format is applied.
PgfBoxColor	<i>Color</i>	Denotes the background color of the paragraph box that encloses the paragraph text.
Direction	int	Set or get the direction of the document. Possible values: <ul style="list-style-type: none"> <li><i>Constants.FV_DIR_Inherit</i> - Inherit the direction of the parent</li> <li><i>Constants.FV_DIR_LTR</i> - <i>Left-to-right</i></li> <li><i>FV_DIR_RTL</i> - <i>Right-to-left</i></li> </ul>

Property name	Data Type	Description
ResolvedDirection	int	Get the inherited direction of the document. Possible values: <ul style="list-style-type: none"> <li><i>Constants.FV_DIR_LTR</i> - Left-to-right</li> <li><i>FV_DIR_RTL</i> - Right-to-left</li> </ul>
AsianComposer	bool	Whether the document will use the Asian Composer layout engine.
UserString	string	A string to which clients can store private data.

## Pgf methods

[Delete](#), [GetProps](#), [GetText](#), [ObjectValid](#), [SetProps](#).

## PgfFmt

Property name	Data Type	Description
AcrobatLevel	int	Retained in Version 6.0 or later for backward compatibility. Use PDFStructureLevel instead.
PgfMarkedForNamedDestination	int	If 1, this paragraph will have a corresponding Named Destination in the generated PDF.
PDFStructureLevel	int	The PDF structure level of paragraphs with the current format. This property is used when the PDFStructure property is True for the document, and FrameMaker generates PDF data. The value for this property can be between 0 and 100, where greater values are deeper in the hierarchy. If PDFStructureLevel is 0, FrameMaker does not include paragraphs of this format in the PDF structure.
MinJRomSpace	int	Minimum Asian-Roman space.
OptJRomSpace	int	Optimum Asian-Roman space.

Property name	Data Type	Description
MaxJRomSpace	int	Maximum Asian-Roman space.
MinJLetSpace	int	Minimum Asian letter space.
OptJLetSpace	int	Optimum Asian letter space.
MaxJLetSpace	int	Maximum Asian letter space.
YakumonoType	int	The Yakumono rules to handle punctuation characters. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_FLOATING_YAKUMONO (0x00)</li> <li>Constants.FV_MONOSPACE_YAKUMONO (0x01)</li> <li>Constants.FV_FIXED_YAKUMONO (0x02)</li> </ul>
AutoNumChar	string	Character format for the automatic numbering string specified by AutoNumString; "" if the default character format is used
AutoNumString	string	Autonumber format string (for example, <n>.<n+>)
NumAtEnd	int	1 if numbering position is End of Paragraph; 0 if it is Beginning of Paragraph.
PgflsAutoNum	int	1 if autonumbering is enabled.
Capitalization	int	Type of capitalization to use. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAPITAL_CASE_NORM (0)</li> <li>Constants.FV_CAPITAL_CASE_SMALL (1)</li> <li>Constants.FV_CAPITAL_CASE_LOWER (2)</li> <li>Constants.FV_CAPITAL_CASE_UPPER (3)</li> </ul>
ChangeBar	int	1 if Change Bars are on.
Color	<i>Color</i>	Spot color (FO_Color ID).

Property name	Data Type	Description
CombinedFontFamily	<i>CombinedFontDefn</i>	Combined font definition (FO_CombinedFontDefn).
FontAngle	int	Font angle (specifies an index into the array of font angles provided by the session property <code>FontAngleNames</code> ).
FontEncodingName	string	The font's encoding.
FontFamily	int	Font family (specifies an index into the array of font families provided by the session property <code>FontFamilyNames</code> ).
FontPlatformName	string	Name that uniquely identifies a font on a specific platform.
FontPostScriptName	string	Name given to a font when it is sent to a PostScript printer.
FontSize	int	Font size (2 pt to 400 pt).
FontVariation	int	Font variation (specifies an index into the array of font variations provided by the session property <code>FontVariationNames</code> ).
FontWeight	int	Font weight (specifies an index into the array of font weights provided by the session property <code>FontWeightNames</code> ).
KernX	int	Horizontal kern value for manual kerning expressed as a percentage of an em (metric –100% to 1000%). A positive value moves a character right and a negative value moves a character left.
KernY	int	Vertical kern value for manual kerning expressed as a percentage of an em (metric –100% to 1000%). A positive value moves characters up and a negative value moves characters down.
Overline	int	1 if Overline is enabled.
PairKern	int	1 if Pair Kern is enabled.

Property name	Data Type	Description
Position	int	Specifies position relative to baseline of text. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_POS_NORM (0): Normal</li> <li>Constants.FV_POS_SUPER (1): Superscript</li> <li>Constants.FV_POS_SUB (2): Subscript</li> </ul>
Stretch	int	Character stretch (set width) expressed as a percentage of normal stretch for the font (metric –10% to 1000%).
Strikethrough	int	1 if Strikethrough is enabled.
Underlining	int	Type of underlining. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CB_NO_UNDERLINE (0)</li> <li>Constants.FV_CB_SINGLE_UNDERLINE (1)</li> <li>Constants.FV_CB_DOUBLE_UNDERLINE (2)</li> <li>Constants.FV_CB_NUMERIC_UNDERLINE (3)</li> </ul>
AdjHyphens	int	Number of allowable adjacent hyphens
Hyphenate	int	1 if Automatic Hyphenation is enabled.
HyphMinPrefix	int	Minimum number of letters that must precede hyphen.
HyphMinSuffix	int	Minimum number of letters that must follow a hyphen.
HyphMinWord	int	Minimum length of a hyphenated word.



Property name	Data Type	Description
Language	int	<p>Hyphenation and spell-checking language to use. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li>Constants.FV_LANG_ENGLISH (0x01)</li> <li>Constants.FV_LANG_BRITISH (0x02)</li> <li>Constants.FV_LANG_GERMAN (0x03)</li> <li>Constants.FV_LANG_SWISS_GERMAN (0x04)</li> <li>Constants.FV_LANG_FRENCH (0x05)</li> <li>Constants.FV_LANG_CANADIAN_FRENCH (0x06)</li> <li>Constants.FV_LANG_SPANISH (0x07)</li> <li>Constants.FV_LANG_CATALAN (0x08)</li> <li>Constants.FV_LANG_ITALIAN (0x09)</li> <li>Constants.FV_LANG_FINNISH (0x10)</li> <li>Constants.FV_LANG_SWEDISH (0x11)</li> <li>Constants.FV_LANG_JAPANESE (0x12)</li> <li>Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>Constants.FV_LANG_KOREAN (0x15)</li> <li>Constants.FV_LANG_PORTUGUESE (0x0A)</li> <li>Constants.FV_LANG_BRAZILIAN (0x0B)</li> <li>Constants.FV_LANG_DANISH (0x0C)</li> <li>Constants.FV_LANG_DUTCH (0x0D)</li> </ul>
		<p>Constants.FV_LANG_NORWEGIAN (0x0E)</p> <p>Constants.FV_LANG_NYNORSK</p>

Property name	Data Type	Description
FirstIndent	int	First-line left margin, measured from left side of current text column (0 cm to 100 cm).
LeftIndent	int	Left margin, measured from left side of current text column (0 cm to 100 cm).
RightIndent	int	Right margin, measured from right side of current text column.
LastLineRightIndent	int	Right margin of the last line in the paragraph, measured from right side of the current text column.
Leading	int	Space below each line in a paragraph.
LineSpacing	int	Space between lines in a paragraph measured from baseline to baseline. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_FIXED (0x00): default font size</li> <li>Constants.FV_PGF_PROPORTIONAL (0x01): largest font in line</li> <li>Constants.FV_PGF_FLOATING (0x02): largest ascender in line</li> </ul>
NextPgInDoc	<i>Pgf</i>	Next paragraph in the document (FO_Pgf ID).
BlockLines	int	The number of Widow/Orphan lines.
KeepWithNext	int	1 if Keep With Next Paragraph is enabled.
KeepWithPrev	int	1 if Keep With Previous Paragraph is enabled.
PgfAlignment	int	Horizontal alignment of paragraph. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_LEFT (0x01)</li> <li>Constants.FV_PGF_RIGHT (0x02)</li> <li>Constants.FV_PGF_CENTER (0x03)</li> <li>Constants.FV_PGF_JUSTIFIED (0x04)</li> </ul>

Property name	Data Type	Description
Placement	int	Paragraph placement. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_SIDE BODY (0) <i>Constants.FV_PGF_SIDE-HEAD_TOP (1)</i> <i>Constants.FV_PGF_SIDEHEAD_ -FIRST_BASELINE (2)</i> <i>Constants.FV_PGF_SIDE-HEAD_LAST_BASELINE (3)</i> <i>Constants.FV_PGF_RUN_IN (4)</i> <i>Constants.FV_PGF_STRADDLE (5)</i> <i>Constants.FV_PGF_STRAD- DLE_NORMAL_ONLY (6)</i></li> </ul>
RunInSeparator	string	String for Run-In Head Default Punctuation.
SpaceAbove	int	Space above paragraph.
SpaceBelow	int	Space below paragraph.
Start	int	Vertical placement of paragraph. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_ANYWHERE (0x00) <i>Constants.FV_PGF_TOP_OF_COL (0x01)</i> <i>Constants.FV_PG- F_TOP_OF_PAGE (0x02)</i> <i>Constants.FV_PG- F_TOP_OF_LEFT_PAGE (0x03)</i> <i>Constants.FV_PG- F_TOP_OF_RIGHT_PAGE (0x04)</i></li> </ul>
BottomSeparator	string	Name of frame to put below paragraph.
TopSeparator	string	Name of frame to put above paragraph.

Property name	Data Type	Description
BkColor	<i>Color</i>	Denotes the text background color. If this property is applied on a paragraph format, it denotes the background color of the whole paragraph; if it is applied on a character format, it denotes the background color of the text over which the character format is applied.
PgfBoxColor	<i>Color</i>	Denotes the background color of the paragraph box that encloses the paragraph text.
UseBkColor	bool	The UseBkColor is a Boolean flag which indicates whether the BkColor property is set or not. A value of 1 indicates the text has some background color. A value of 0 indicates a transparent background. <i>NOTE: When the UseBkColor property is set to 0, the BkColor value is not used.</i>
CellBottomMargin	int	Amount added to default bottom margin of table cell.
CellLeftMargin	int	Amount added to default left margin of table cell.

Property name	Data Type	Description
CellMarginsFixed	int	<p>Specifies which cell margins are fixed. The following values can be ORed into it:</p> <ul style="list-style-type: none"> <li>• <i>Constants.FV_PGF_FIXED_L_MARGIN (0x01): the left margin is fixed.</i></li> <li>• <i>Constants.FV_PGF_FIXED_B_MARGIN (0x02): the bottom margin is fixed.</i></li> <li>• <i>Constants.FV_PGF_FIXED_R_MARGIN (0x04): the right margin is fixed.</i></li> <li>• <i>Constants.FV_PGF_FIXED_T_MARGIN (0x08): the top margin is fixed.</i></li> </ul> <p>If the margin for a cell is fixed, the margin property specifies the absolute value of the cell margin. For example, if <i>Constants.FV_PGF_FIXED_B_MARGIN (0x02)</i> is set, <i>CellBottomMargin</i> specifies the absolute value of the cell's bottom margin, overriding the cell margin specified by the table format.</p> <p>If <i>Constants.FV_PGF_FIXED_B_MARGIN (0x02)</i> is not set, <i>CellBottomMargin</i> is added to the margin specified by the table format.</p>
CellRightMargin	int	Amount added to default right margin of table cell.
CellTopMargin	int	Amount added to default top margin of table cell.

Property name	Data Type	Description
CellVAlignment	int	Vertical alignment of a paragraph when it is the first one in a cell. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_V_ALIGN_TOP (0)</li> <li>Constants.FV_PGF_V_ALIGN_MIDDLE (1)</li> <li>Constants.FV_PGF_V_ALIGN_BOTTOM (2)</li> </ul>
NumTabs	int	Number of tabs in the paragraph..
Tabs	<i>Tabs</i>	Array of tab descriptions that specify the positions and types of tab stops.
Name	string	Name of paragraph format.
NextTag	string	Tag for new next paragraph.
UseNextTag	int	1 if Next Paragraph Tag is enabled.
LetterSpace	int	1 if Word Spacing is enabled.
MaxSpace	int	Maximum word spacing (percentage of an em space in current font).
MinSpace	int	Minimum word spacing (percentage of an em space in current font).
OptSpace	int	Optimum word spacing.
Direction	int	Set or get the direction of the document. Possible values: <ul style="list-style-type: none"> <li>Constants.FV_DIR_Inherit - Inherit the direction of the parent</li> <li>Constants.FV_DIR_LTR - Left-to-right</li> <li>FV_DIR_RTL - Right-to-left</li> </ul>
ResolvedDirection	int	Get the inherited direction of the document. Possible values: <ul style="list-style-type: none"> <li>Constants.FV_DIR_LTR - Left-to-right</li> <li>FV_DIR_RTL - Right-to-left</li> </ul>
AsianComposer	bool	Whether the document will use the Asian Composer layout engine.

## PgfFmt methods

*Delete*, *GetProps*, *ObjectValid*, *SetProps*.

## Point

Property name	Data Type	Description
x	int	X coordinate
y	int	Y coordnate

## Point methods

*Point*.

## Points

An Array of *Point* objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## Points methods

*concat*, *pop*, *push*.

## Polygon

Property name	Data Type	Description
NumPoints	int	Number of polygon vertices.
Points	<i>Points</i>	Array of x-y coordinate pairs that specify the polygon's vertices.
PolyIsBezier	int	1 if polygon is smoothed.

Property name	Data Type	Description
Angle	<i>int</i>	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	<i>int</i>	Denotes the arrowhead base angle in degrees.
ArrowLength	<i>int</i>	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	<i>int</i>	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is <code>False</code> .
ArrowScaleHead	<i>int</i>	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	<i>int</i>	Denotes the arrowhead tip angle in degrees.
ArrowType	<i>int</i>	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>• <code>Constants.FV_ARROW_STICK (0x1)</code> <i>Constants.FV_ARROW_HOLLOW (0x2)</i> <i>Constants.FV_ARROW_FILLED (0x3)</i></li> </ul>
BorderWidth	<i>int</i>	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	<i>Color</i>	Denotes the spot color.
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.



Property name	Data Type	Description
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FMOBJECT	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	<i>Group</i>	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.

Property name	Data Type	Description
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is –216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is –216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FMOBJECT	Denotes the next graphic object in the document.
NextGraphicInFrame	FMOBJECT	Denotes the next graphic object in the frame.
NextGraphicInGroup	FMOBJECT	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FMOBJECT	Denotes the next selected graphic object in the document.

Property name	Data Type	Description
ObjectAttributes	<i>Strings</i>	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.

Property name	Data Type	Description
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Indicates the graphic object's internal unique number.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## Polygon methods

*Delete, GetProps, ObjectValid, SetProps.*

## Polyline

Property name	Data Type	Description
NumPoints	int	Number of polygon vertices.
Points	<i>Points</i>	Array of x-y coordinate pairs that specify the polygon's vertices.
PolyIsBezier	int	1 if polygon is smoothed.
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is False.

Property name	Data Type	Description
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	<i>Color</i>	Denotes the spot color.
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FMOBJECT	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.

Property name	Data Type	Description
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	<i>Group</i>	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.

Property name	Data Type	Description
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is –216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FObject	Denotes the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Denotes the next selected graphic object in the document.
ObjectAttributes	<i>Strings</i>	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>

Property name	Data Type	Description
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Indicates the graphic object's internal unique number.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## Polyline methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).



## PropIdent

Property name	Data Type	Description
num	int	The property number.
name	string	The property name.

### PropIdent methods

*PropIdent*.

## PropVal

Property name	Data Type	Description
propIdent	<i>PropIdent</i>	The property identifier.
propVal	<i>TypedVal</i>	The property value.

### PropVal methods

*PropVal*.

## PropVals

An Array of *PropVal* objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

### PropVals methods

*concat*, *pop*, *push*.

## Rectangle

Property name	Data Type	Description
RectangleIsSmoothed	int	1 if smoothing is enabled. This property is always 1 for Ellipse objects.
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>• <code>Constants.FV_ARROW_STICK (0x1)</code> <i>Constants.FV_ARROW_HOLLOW (0x2)</i> <i>Constants.FV_ARROW_FILLED (0x3)</i></li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	<i>Color</i>	Denotes the spot color.

Property name	Data Type	Description
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	<i>Group</i>	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.

Property name	Data Type	Description
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FMOBJECT	Denotes the next graphic object in the document.
NextGraphicInFrame	FMOBJECT	Denotes the next graphic object in the frame.
NextGraphicInGroup	FMOBJECT	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FMOBJECT	Denotes the next selected graphic object in the document.

Property name	Data Type	Description
ObjectAttributes	<i>Strings</i>	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.

Property name	Data Type	Description
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Indicates the graphic object's internal unique number.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## Rectangle methods

*Delete, GetProps, ObjectValid, SetProps.*

## RefPage

Property name	Data Type	Description
Name	string	Name of the reference page.
PageFrame	<i>UnanchoredFrame</i>	Page frame (FO_UnanchoredFrame ID)
PageHeight	int	Height of the page.
PageNext	<i>RefPage</i>	Next reference page (FO_RefPage ID) in the document.
PageNum	int	Page number.
PagePrev	<i>RefPage</i>	Previous reference page (FO_RefPage ID) in the document.
PageWidth	int	Width of the page.

## RefPage methods

*ApplyPageLayout, Delete, GetProps, ObjectValid, SetProps.*

## RoundRect

Property name	Data Type	Description
Radius	int	Radius of corner; 0 for a square corner.
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>• <code>Constants.FV_ARROW_STICK (0x1)</code> <i>Constants.FV_ARROW_HOLLOW (0x2)</i> <i>Constants.FV_ARROW_FILLED (0x3)</i></li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	<i>Color</i>	Denotes the spot color.

Property name	Data Type	Description
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	<i>Group</i>	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.



Property name	Data Type	Description
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FObject	Denotes the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Denotes the next selected graphic object in the document.

Property name	Data Type	Description
ObjectAttributes	<i>Strings</i>	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.

Property name	Data Type	Description
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Indicates the graphic object's internal unique number.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## RoundRect methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## Row

Property name	Data Type	Description
CondFmtIsShown	int	1 if the condition is shown.
Element	<a href="#">Element</a>	The ID of the element containing the row in a FrameMaker document.
FirstCellInRow	<a href="#">Cell</a>	First cell in row (FO_Cell ID).
Height	int	Height of the row.
InCond	Objects	Condition tags for row (array of FO_CondFmt IDs).
LocX	int	Offset from the left side of the text frame containing the row.
LocY	int	Offset from the top of the page frame containing the row.
NextRowInTbl	<a href="#">Row</a>	Next row (FO_Row ID) in the table.
PrevRowInTbl	<a href="#">Row</a>	Previous row (FO_Row ID) in the table.
RowsShown	int	1 if the conditional row is shown.
RowKeepWithNext	int	1 if Keep With Next Row is enabled.

Property name	Data Type	Description
RowKeepWithPrev	int	1 if Keep With Previous Row is enabled.
RowMaxHeight	int	Maximum row height.
RowMinHeight	int	Minimum row height.
RowStart	int	Denotes the row placement. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ROW_ANYWHERE (0)</li> <li>Constants.FV_ROW_TOP_OF_COL (1)</li> <li>Constants.FV_ROW_TOP_OF_PAGE (2)</li> <li>Constants.FV_ROW_TOP_OF_LEFT_PAGE (3)</li> <li>Constants.FV_ROW_TOP_OF_RIGHT_PAGE (4)</li> </ul>
RowTbl	<i>Tbl</i>	Table containing the row (FO_Tbl ID).
RowType	int	Type of row. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ROW_HEADING (0)</li> <li>Constants.FV_ROW_BODY (1)</li> <li>Constants.FV_ROW_FOOTING (2)</li> </ul>
SepOverride	<i>Color</i>	Color separation format override (FO_Color ID).

Property name	Data Type	Description
StyleOverrides	int	<p>Style condition indicators for conditional text:</p> <ul style="list-style-type: none"> <li>Constants.FV_CS_NO_OVERRIDE (0x00)</li> <li>Constants.FV_CS_OVERLINE (0x01)</li> <li>Constants.FV_CS_STRIKE-THROUGH (0x02)</li> <li>Constants.FV_CS_SINGLE_UNDERLINE (0x04)</li> <li>Constants.FV_CS_DOUBLE_UNDERLINE (0x08)</li> </ul> <p>All style condition indicators are represented as hatched lines for the table rows.</p>
UseSepOverride	int	1 if the SepOverride property overrides default from the table.
Width	int	Width of the row.

## Row methods

[AddRows](#), [Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## Rubi

Property name	Data Type	Description
Element	<i>Element</i>	The object handle of the associated FO_Element for the rubi group element, if the rubi group is in a structured document.
OyamojiTextRange	<i>TextRange</i>	The text range that the oyamoji text encompasses.
NextRubiInDoc	<i>Rubi</i>	The next instance of a rubi composite (FO_Rubi ID) in the document..

Property name	Data Type	Description
RubiElement	<i>Element</i>	The object handle of the associated FO_Element for the rubi element, if the rubi group is in a structured document.
RubiTextRange	<i>TextRange</i>	The text range that the rubi text encompasses.
Unique	int	The rubi composite's unique ID.

## Rubi methods

*Delete*, *GetProps*, *GetText2*, *ObjectValid*, *SetProps*.

## RulingFmt

Property name	Data Type	Description
Name	string	Ruling format name.
NextRulingFmtInDoc	<i>RulingFmt</i>	Next ruling format in document (FO_RulingFmt ID).
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
RulingGap	int	Gap between double ruling lines (0.015 pt to 360 pt).
RulingLines	int	Number of ruling lines (0 to 2 lines).
RulingPenWidth	int	Ruling line thickness (0.015 pt to 360 pt).
RulingSep	<i>Color</i>	Spot color of ruling format (FO_Color ID).

## RulingFmt methods

[Delete](#), [GetProps](#), [ObjectValid](#), [SetProps](#).

## Strings

An Array of objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## Strings methods

[concat](#), [pop](#), [push](#).

## SubCol

Property name	Data Type	Description
ContentHeight	int	The distance between the top of the column and the baseline of the last line in the column.
FirstAFrame	<a href="#">AFrame</a>	First anchored frame in the column (FO_AFrame ID).
FirstCell	<a href="#">Cell</a>	First table cell in the column (FO_Cell ID).
FirstFn	<a href="#">Fn</a>	First footnote in the column (FO_Fn ID).
FirstPgf	<a href="#">Pgf</a>	First paragraph in the column (FO_Pgf ID).
FrameParent	<a href="#">TextFrame</a>	ID of text frame that contains the column (FO_TextFrame ID).
Height	int	Column Height.
LastAFrame	<a href="#">AFrame</a>	Last anchored frame in the column (FO_AFrame ID).
LastCell	<a href="#">Cell</a>	Last table cell in the column (FO_Cell ID).

Property name	Data Type	Description
LastFn	<i>Fn</i>	Last footnote in the column (FO_Fn ID).
LastPgf	<i>Pgf</i>	Last paragraph in the column (FO_Pgf ID).
LocX	int	Offset from left side of the text frame that contains the column.
LocY	int	Offset from top of text frame that contains the column.
NextSubCol	<i>SubCol</i>	Next column in the flow.
Overflowed	int	1 if the text frame containing the column has Autoconnect turned off and text overflows the column.
ParentTextFrame	<i>TextFrame</i>	ID of text frame that contains the column (FO_TextFrame ID).
PrevSubCol	<i>SubCol</i>	Previous column in the flow.
Unique	int	Text column's unique ID.
UserString	string	A string to which clients can store private data.
Width	int	Column width.

## SubCol methods

*Delete, GetProps, GetText, ObjectValid, SetProps.*

## Tab

Property name	Data Type	Description
x	int	Offset from the left margin.



Property name	Data Type	Description
type	uint	Type of tab. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TAB_LEFT (0x1) - Left tab.</li> <li>Constants.FV_TAB_CENTER (0x2) - Center tab.</li> <li>Constants.FV_TAB_RIGHT (0x3) - Right tab.</li> <li>Constants.FV_TAB_DECIMAL (0x4) - Decimal tab.</li> <li>Constants.FV_TAB_RELATIVE_LEFT (0x5) - Relative left tab (allowed only for format change lists).</li> <li>Constants.FV_TAB_RELATIVE_CENTER (0x6) - Relative center tab (allowed only for format change lists).</li> <li>Constants.FV_TAB_RELATIVE_RIGHT (0x7) - Relative right tab (allowed only for format change lists).</li> <li>Constants.FV_TAB_RELATIVE_DECIMAL (0x8) - Relative decimal tab (allowed only for format change lists).</li> </ul>
leader	string	String that appears before the tab.
decimal	uint	Character to align the tab around (For example ` , `”).

## Tab methods

[Tab](#).

## Tabs

An Array of [Tab](#) objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## Tabs methods

*concat, pop, push.*

## Tbl

Property name	Data Type	Description
ContentHeight	int	The height of the table title.
Locked	int	1 if the table is part of a text inset that retains formatting information from the source document. The table is not affected by global formatting performed on the document.
OrphanRows	int	Number of orphan rows.
Overflowed	int	1 if the table has cells that are not shown because they extend beyond the text frame boundaries.
TblAlignment	int	Horizontal placement of table. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ALIGN_TBL_LEFT (0)</li> <li>Constants.FV_ALIGN_TBL_CENTER (1)</li> <li>Constants.FV_ALIGN_TBL_RIGHT (2)</li> </ul>
TblCellBottomMargin	int	Default bottom cell margin for the table.
TblCellLeftMargin	int	Default left cell margin for the table.
TblCellRightMargin	int	Default right cell margin for the table.
TblCellTopMargin	int	Default top cell margin for the table.
TblLeftIndent	int	Left indent for the table.

Property name	Data Type	Description
TblInLockedTi	bool	1 if the table is in a locket text inset. You should not use the ESTK to delete table elements when the table is in a locket text inset.
TblPlacement	int	Vertical placement of table on page. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TBL_ANYWHERE (0)</li> <li>Constants.FV_TBL_TOP_OF_COL (1)</li> <li>Constants.FV_TBL_TOP_OF_PAGE (2)</li> <li>Constants.FV_TBL_TOP_OF_LEFT_PAGE (3)</li> <li>Constants.FV_TBL_TOP_OF_RIGHT_PAGE (4)</li> <li>Constants.FV_TBL_FLOAT (5)</li> </ul>
TblRightIndent	int	Right indent for the table.
TblSpaceAbove	int	Vertical space above the table.
TblSpaceBelow	int	Vertical space below the table.
TextLoc	<i>TextLoc</i>	The text location of the table's anchor.
Unique	int	The table's unique ID.
FirstRowInTbl	<i>Row</i>	First row in the table (FO_Row ID).
LastRowInTbl	<i>Row</i>	Last row in the table (FO_Row ID).
NextTblInDoc	<i>Tbl</i>	Next table (FO_Tbl ID) in the document.
TblCatalogEntry	int	1 if the table's format is in the Table Catalog.
TblColWidths	<i>Metrics</i>	List of column widths.
TblNumbering	int	Direction of autonumbering for the table. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TBL_NUM_BY_ROW (0)</li> <li>Constants.FV_TBL_NUM_BY_COL (1)</li> </ul>
TblNumCols	int	Number of columns in the table.

Property name	Data Type	Description
TblNumRows	int	Number of rows in the table.
TblTag	string	Name of the table format.
TblWidth	int	Horizontal width of the table.
TblBodyRowRuling	<i>RulingFmt</i>	Ruling applied to body rows specified by TblBodyRowRulingPeriod (FO_RulingFmt ID).
TblBodyRowRulingPeriod	int	The periodicity of the ruling specified by TblBodyRowRuling. For example, if TblBodyRowRulingPeriod is set to 3, the ruling specified by TblBodyRowRuling is applied to every third row.
TblBottomRuling	int	Ruling applied to the bottom of the table (FO_RulingFmt ID).
TblColRuling	<i>RulingFmt</i>	Ruling applied to table columns specified by TblColRulingPeriod (FO_RulingFmt ID).
TblColRulingPeriod	int	The periodicity of the ruling specified by TblColRuling. For example, if TblColRulingPeriod is set to 2, the ruling specified by TblColRuling is applied to every other column.
TblHFRowRuling	<i>RulingFmt</i>	Ruling for table heading and footing rows (FO_RulingFmt ID).
TblHFSeparatorRuling	<i>RulingFmt</i>	Separator ruling for table heading and footing rows (FO_RulingFmt ID).
TblLastBodyRuling	int	1 if Draw Bottom Ruling on Last Sheet Only is enabled (FO_RulingFmt ID).
TblLeftRuling	<i>RulingFmt</i>	Ruling for the left side of the table (FO_RulingFmt ID).
TblOtherBodyRowRuling	<i>RulingFmt</i>	Ruling for body rows that are not specified by TblBodyRowRulingPeriod (FO_RulingFmt ID).

Property name	Data Type	Description
TblOtherColRuling	<i>RulingFmt</i>	Ruling for columns that are not specified by TblColRulingPeriod (FO_RulingFmt ID).
TblRightRuling	<i>RulingFmt</i>	Ruling for the right side of the table (FO_RulingFmt ID).
TblTopRuling	<i>RulingFmt</i>	Ruling for the top of the table (FO_RulingFmt ID).
BottomRowSelection	<i>Row</i>	Bottom body row in selection, if table is selected (FO_Row ID).
LeftColNum	int	Number of the leftmost selected column, if a table is selected (columns are numbered from left to right, starting with 0).
RightColNum	int	Number of rightmost selected column, if a table is selected (columns are numbered from left to right, starting with 0).
TblTitleSelected	int	1 if table title is selected.
TopRowSelection	<i>Row</i>	Top row in selection, if table is selected (FO_Row ID).
TblBodyFirstColor	<i>Color</i>	First spot color for table body (FO_Color ID).
TblBodyFirstFill	int	First fill pattern for table body.
TblBodyFirstPeriod	int	Number of columns or body rows to which the first fill pattern (specified by TblBodyFirstFill) is applied.
TblBodyNextColor	<i>Color</i>	Exception color for columns or body rows (FO_Color ID).
TblBodyNextFill	int	Exception fill pattern for table body.
TblBodyNextPeriod	int	Number of columns or body rows to which the exception fill pattern (specified by TblBodyNextFill) is applied.
TblBodyShadeBy	int	1 if Shade By is set to Columns; 0 if Shade By is set to Rows.

Property name	Data Type	Description
TblHFColor	<i>Color</i>	Color for table heading and footing.
TblHFFill	int	Fill pattern for table heading and footing (integer percentage).
Element	<i>Element</i>	Denotes the element associated with the table.
TblBodyElement	<i>Element</i>	Specifies the element containing the table's body rows.
TblElement	<i>Element</i>	Specifies the element containing the table.
TblFooterElement	<i>Element</i>	Denotes the element containing the table's footer rows.
TblHeaderElement	<i>Element</i>	Denotes the element containing the table's header rows.
TblTitleElement	<i>Element</i>	Indicates the element containing the table title.
FirstPgf	<i>Pgf</i>	The first paragraph in the title, if the table has a title.
HighestLevelElement	<i>Element</i>	Specifies the title's highest-level element if the table is in a structured document and has a title. <b>NOTE:</b> <i>HighestLevelElement</i> is obsolete but is supported for backward compatibility.
LastPgf	<i>Pgf</i>	The last paragraph in the title, if the table has a title.
TblTitleGap	int	Gap between the title and top or bottom row of the table.

Property name	Data Type	Description
TblTitlePosition	int	The placement of the table title. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TBL_NO_TITLE (0) - Table has no title</li> <li>Constants.FV_TBL_TITLE_ABOVE (1) - The title appears above the table</li> <li>Constants.FV_TBL_TITLE_BELOW (2) - The title appears below the table</li> </ul>
Direction	int	Set or get the direction of the document. Possible values: <ul style="list-style-type: none"> <li>Constants.FV_DIR_Inherit - Inherit the direction of the parent</li> <li>Constants.FV_DIR_LTR - Left-to-right</li> <li>FV_DIR_RTL - Right-to-left</li> </ul>
ResolvedDirection	int	Get the inherited direction of the document. Possible values: <ul style="list-style-type: none"> <li>Constants.FV_DIR_LTR - Left-to-right</li> <li>FV_DIR_RTL - Right-to-left</li> </ul>
UserString	string	A string to which clients can store private data.

## Tbl methods

[AddCols](#), [Delete](#), [DeleteCols](#), [DeleteRows](#), [GetTableColumnByColNum](#), [GetProps](#), [MakeTblSelection](#), [ObjectValid](#), [SetProps](#).

## TblFmt

Property name	Data Type	Description
OrphanRows	int	Number of orphan rows.

Property name	Data Type	Description
TblAlignment	int	Horizontal placement of table. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ALIGN_TBL_LEFT (0)</li> <li>Constants.FV_ALIGN_TBL_CENTER (1)</li> <li>Constants.FV_ALIGN_TBL_RIGHT (2)</li> </ul>
TblCellBottomMargin	int	Bottom margin of table cell.
TblCellLeftMargin	int	Left margin of the table cell.
TblCellRightMargin	int	Right margin of the table cell.
TblCellTopMargin	int	Top margin of the table cell.
TblLeftIndent	int	Left indent of the table cell.
TblPlacement	int	Vertical placement of table on page. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TBL_ANYWHERE (0)</li> <li>Constants.FV_TBL_TOP_OF_COL (1)</li> <li>Constants.FV_TBL_TOP_OF_PAGE (2)</li> <li>Constants.FV_TBL_TOP_OF_LEFT_PAGE (3)</li> <li>Constants.FV_TBL_TOP_OF_RIGHT_PAGE (4)</li> <li>Constants.FV_TBL_FLOAT (5)</li> </ul>
TblRightIndent	int	Right indent of the table.
TblSpaceAbove	int	Vertical space above the table.
TblSpaceBelow	int	Vertical space below the table.
TblTitleGap	int	Gap between title and top or bottom row.



Property name	Data Type	Description
TblTitlePosition	int	The placement of the table title. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TBL_NO_TITLE (0) - Table has no title</li> <li>Constants.FV_TBL_TITLE_ABOVE (1) - The title appears above the table</li> <li>Constants.FV_TBL_TITLE_BELOW (2) - The title appears below the table</li> </ul>
TblCatalogEntry	int	1 if format is in the Table Catalog.
Name	string	Name of the paragraph format of the table title.
NextTblFmtInDoc	<i>TblFmt</i>	Next table format in the document (FO_TblFmt ID).
TblNumbering	int	Direction of autonumbering for the table. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TBL_NUM_BY_ROW (0)</li> <li>Constants.FV_TBL_NUM_BY_COL (1)</li> </ul>
TblTag	string	Name of the table format
TblInitNumBodyRows	int	Number of body rows for new table
TblInitNumCols	int	Number of columns for new table
TblInitNumFRows	int	Number of footing rows for new table
TblInitNumHRows	int	Number of heading rows for new table
TblBodyRowRuling	<i>RulingFmt</i>	Ruling for body rows that are not specified by TblBodyRowRulingPeriod (FO_RulingFmt ID).

Property name	Data Type	Description
TblBodyRowRulingPeriod	int	The periodicity of the ruling specified by TblOtherBodyRowRuling. For example, if TblBodyRowRulingPeriod is set to 3, the ruling specified by TblOtherBodyRowRuling is applied to every third row.
TblBottomRuling	<i>RulingFmt</i>	Ruling for the bottom of the table (FO_RulingFmt ID).
TblColRuling	<i>RulingFmt</i>	Ruling for columns that are not specified by TblColRulingPeriod (FO_RulingFmt ID).
TblColRulingPeriod	int	The periodicity of the ruling specified by TblOtherColRuling. For example, if TblColRulingPeriod is set to 2, the ruling specified by TblOtherColRuling is applied to every other column.
TblHFRowRuling	<i>RulingFmt</i>	Ruling for the heading and footing rows (FO_RulingFmt ID).
TblHFSeparatorRuling	<i>RulingFmt</i>	Separator ruling for the table heading and footing rows (FO_RulingFmtID).
TblLastBodyRuling	int	1 if Draw Bottom Ruling on Last Sheet Only is enabled (FO_RulingFmt ID).
TblLeftRuling	<i>RulingFmt</i>	Ruling for the left side of the table (FO_RulingFmt ID).
TblOtherBodyRowRuling	<i>RulingFmt</i>	Ruling applied to body rows specified by TblBodyRowRulingPeriod (FO_RulingFmt ID).
TblOtherColRuling	<i>RulingFmt</i>	Ruling applied to table columns specified by TblColRulingPeriod (FO_RulingFmt ID).
TblRightRuling	<i>RulingFmt</i>	Ruling for the right side of the table (FO_RulingFmt ID).
TblTopRuling	<i>RulingFmt</i>	Ruling for the top of the table (FO_RulingFmt ID).

Property name	Data Type	Description
TblBodyFirstColor	<i>Color</i>	First spot color for table body (FO_Color ID)
TblBodyFirstFill	int	First fill pattern for table body
TblBodyFirstPeriod	int	Number of columns or body rows to which the first fill pattern (specified by TblBodyFirstFill) is applied
TblBodyNextColor	<i>Color</i>	Exception color for columns or body rows (FO_Color ID)
TblBodyNextFill	int	Exception fill pattern for table body
TblBodyNextPeriod	int	Number of columns or body rows to which the exception fill pattern (specified by TblBodyNextFill) is applied
TblBodyShadeBy	int	1 if Shade By is set to Columns; 0 if Shade By is set to Rows
TblHFColor	<i>Color</i>	Color for table heading and footing
TblHFFill	int	Fill pattern for table heading and footing (integer percentage)
Direction	int	Set or get the direction of the document. Possible values: <ul style="list-style-type: none"> <li><i>Constants.FV_DIR_Inherit</i> - Inherit the direction of the parent</li> <li><i>Constants.FV_DIR_LTR</i> - Left-to-right</li> <li><i>FV_DIR_RTL</i> - Right-to-left</li> </ul>
ResolvedDirection	int	Get the inherited direction of the document. Possible values: <ul style="list-style-type: none"> <li><i>Constants.FV_DIR_LTR</i> - Left-to-right</li> <li><i>FV_DIR_RTL</i> - Right-to-left</li> </ul>
UserString	string	A string to which clients can store private data.

## TblFmt methods

*Delete*, *GetProps*, *ObjectValid*, *SetProps*.

## TextFrame

Property name	Data Type	Description
ColGapWidth	int	Gap between columns (0 to 50 inches).
ColumnsAreBalanced	int	1 if terminal and underfilled columns in the flow are balanced.
FirstAFrame	<i>AFrame</i>	First anchored frame in the text frame (FO_AFrame ID).
FirstCell	<i>Cell</i>	First table cell in the text frame (FO_Cell ID).
FirstFn	<i>Fn</i>	First footnote in the text frame (FO_Fn ID).
FirstPgf	<i>Pgf</i>	First paragraph in the text frame (FO_Pgf ID).
FirstSubCol	<i>SubCol</i>	First column in the text frame (FO_SubCol ID).
Flow	<i>Flow</i>	Flow containing the text frame (FO_Flow ID).
GraphicsButton	int	1 if the text frame is a hypertext button.
LastAFrame	<i>AFrame</i>	Last anchored frame in the text frame (FO_AFrame ID).
LastCell	<i>Cell</i>	Last table cell in the text frame (FO_Cell ID).
LastFn	<i>Fn</i>	Last footnote in the text frame (FO_Fn ID).
LastPgf	<i>Pgf</i>	Last paragraph in the text frame (FO_Pgf ID).
LastSubCol	<i>SubCol</i>	Last column in the text frame (FO_SubCol ID).
NextTextFrameInFlow	<i>TextFrame</i>	Next text frame in the flow (FO_TextFrame ID).
NumColumns	int	The number of columns in the underlying column grid (1–10).

Property name	Data Type	Description
PrevTextFrameInFlow	<i>TextFrame</i>	Previous text frame in the flow (FO_TextFrame ID).
SideHeadGap	int	Gap between side head area and body text area (0 to 50 inches).
SideHeadPlacement	int	Placement of side heads relative to columns in the text frame: <ul style="list-style-type: none"> <li>Constants.FV_SH_LEFT (0x00)</li> <li>Constants.FV_SH_RIGHT (0x01)</li> <li>Constants.FV_SH_INSIDE (0x02)</li> <li>Constants.FV_SH_OUTSIDE (0x03)</li> </ul>
SideHeadWidth	int	Width of side head area for the text frame (0 to 50 inches).
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>

Property name	Data Type	Description
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	<i>Color</i>	Denotes the spot color.
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	<i>Group</i>	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.

Property name	Data Type	Description
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FMOBJECT	Denotes the next graphic object in the document.
NextGraphicInFrame	FMOBJECT	Denotes the next graphic object in the frame.

Property name	Data Type	Description
NextGraphicInGroup	FObject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Denotes the next selected graphic object in the document.
ObjectAttributes	<i>Strings</i>	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.



Property name	Data Type	Description
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Indicates the graphic object's internal unique number.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## TextFrame methods

[Delete](#), [GetProps](#), [GetText](#), [ObjectValid](#), [SetProps](#).

## TextItem

Property name	Data Type	Description
offset	int	Offset from the beginning of the paragraph or the text line.
dataType	int	Type of the text item.
sdata	string	Denotes the text item if the text item is a string.

Property name	Data Type	Description
idata	number	ID of the object if the text item is an object,.
obj	any	Denotes the text item if the text item is an object.

The type of Text Item is one of the following:

Text Item Type	Text Item Represents	Text Item Data
FTI_TextObjId	The object to which the offsets of all the text items are relative.	FO_Pgf, FO_Cell, FO_TextLine, FO_TiApiClient, FO_TiFlow, FO_TiText, or FO_TiTextTable
FTI_String	A string of characters with the same condition and character format.	A character string
FTI_LineBegin	The beginning of a line.	-
FTI_LineEnd	The end of a line and the line end type.	If the line end is a normal line end, the value is 0; if it is a forced line end, the FTI_HardLineEnd flag is set; if it is a hyphen line end, the FTI_HyphenLineEnd flag is set
FTI_PgfBegin	The beginning of a paragraph.	FO_Pgf
FTI_PgfEnd	The end of a paragraph.	FO_Pgf
FTI_FlowBegin	The beginning of a flow.	FO_Flow
FTI_FlowEnd	The end of a flow.	FO_Flow
FTI_PageBegin	The beginning of a page.	FO_Page
FTI_PageEnd	The end of a page.	FO_Page
FTI_SubColBegin	The beginning of a column.	FO_SubCol
FTI_SubColEnd	The end of a column.	FO_SubCol
FTI_FrameAnchor	An anchored frame.	FO_AFrame
FTI_FnAnchor	A footnote.	FO_Fn
FTI_TblAnchor	A table.	FO_Tbl
FTI_MarkerAnchor	A marker.	FO_Marker

Text Item Type	Text Item Represents	Text Item Data
FTI_XRefBegin	The beginning of a cross-reference.	FO_XRef
FTI_XRefEnd	The end of a cross-reference.	FO_XRef
FTI_TextFrameBegin	The beginning of a text frame.	FO_TextFrame
FTI_TextFrameEnd	The end of a text frame.	FO_TextFrame
FTI_VarBegin	The beginning of a variable.	FO_Var
FTI_VarEnd	The end of a variable.	FO_Var
FTI_ElementBegin	The beginning of a container structural element.	FO_Element
FTI_ElementEnd	The end of a container structural element.	FO_Element
FTI_ElemPrefixBegin	The beginning of an element's prefix.	FO_Element
FTI_ElemPrefixEnd	The end of an element's prefix.	FO_Element
FTI_ElemSuffixBegin	The beginning of an element's suffix.	FO_Element
FTI_ElemSuffixEnd	The end of an element's suffix.	FO_Element
FTI_CharPropsChange	A change in the text properties.	Flags indicating which properties have changed (see the following table).
FTI_RubiCompositeBegin	The beginning of a rubi composite (and the beginning of oyamoji text).	FO_Rubi
FTI_RubiCompositeEnd	The end of a rubi composite.	FO_Rubi
FTI_RubiTextBegin	The beginning of rubi text (and the end of oyamoji text).	FO_Rubi
FTI_RubiTextEnd	The end of rubi text.	FO_Rubi

The following table lists the flags for `FTI_CharPropsChange`. These flags indicate which properties have changed.

Flags	Meaning
FTF_ALL	OR of all the flags

Flags	Meaning
FTF_ANGLE	The font angle has changed.
FTF_CAPITALIZATION	The capitalization has changed.
FTF_CHANGEBAR	The change bars have changed.
FTF_CHARTAG	The Character Catalog format has changed.
FTF_COLOR	The color has changed.
FTF_CONDITIONTAG	The condition tag has changed.
FTF_ENCODING	The text encoding has changed,
FTF_FAMILY	The font family has changed.
FTF_IIF	An internal flag having to do with Asian text. input. If there is a non-zero value for this flag, a front-end processor is controlling that text and you should not modify the associated text item.
FTF_KERNX	The kern-x characteristic has changed.
FTF_KERNY	The kern-x characteristic has changed.
FTF_LANGUAGE	Character language has changed.
FTF_OUTLINE	The outline characteristic has changed.
FTF_OVERLINE	The overline characteristic has changed.
FTF_PAIRKERN	The pair kerning has changed.
FTF_POSITION	The character position has changed.
FTF_SHADOW	The shadow characteristic has changed.
FTF_SIZE	The font size has changed.
FTF_SPREAD	The font spread has changed.
FTF_STRETCH	The font stretch value has changed.
FTF_STRIKETHROUGH	The strikethrough characteristic has changed.
FTF_TSUME	The Tsume setting has changed.
FTF_UNDERLINING	The underlining has changed.
FTF_VARIATION	The font variation has changed.
FTF_WEIGHT	The font weight has changed.

## TextItems

An Array of *TextItem* objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

### TextItems methods

*concat*, *pop*, *push*.

## TextLine

Property name	Data Type	Description
BasePointX	int	Horizontal placement of text line base point relative to left side of the frame.
BasePointY	int	Vertical placement of text line base point relative to top of the frame.

Property name	Data Type	Description
Language	int	<p>Hyphenation and Spell Checking language to use.</p> <ul style="list-style-type: none"> <li>Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li>Constants.FV_LANG_ENGLISH (0x01)</li> <li>Constants.FV_LANG_BRITISH (0x02)</li> <li>Constants.FV_LANG_GERMAN (0x03)</li> <li>Constants.FV_LANG_SWISS_GERMAN (0x04)</li> <li>Constants.FV_LANG_FRENCH (0x05)</li> <li>Constants.FV_LANG_CANADIAN_FRENCH (0x06)</li> <li>Constants.FV_LANG_SPANISH (0x07)</li> <li>Constants.FV_LANG_CATALAN (0x08)</li> <li>Constants.FV_LANG_ITALIAN (0x09)</li> <li>Constants.FV_LANG_FINNISH (0x10)</li> <li>Constants.FV_LANG_SWEDISH (0x11)</li> <li>Constants.FV_LANG_JAPANESE (0x12)</li> <li>Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>Constants.FV_LANG_KOREAN (0x15)</li> <li>Constants.FV_LANG_PORTUGUESE (0x0A)</li> <li>Constants.FV_LANG_BRAZILIAN (0x0B)</li> <li>Constants.FV_LANG_DANISH (0x0C)</li> <li>Constants.FV_LANG_DUTCH (0x0D)</li> </ul>
		<ul style="list-style-type: none"> <li>Constants.FV_LANG_NORWEGIAN (0x0E)</li> <li>Constants.FV_LANG_NYNORSK</li> </ul>

Property name	Data Type	Description
TextLineType	int	Type of text line. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TEXTLINE_LEFT (0)</li> <li>Constants.FV_TEXTLINE_RIGHT (1)</li> <li>Constants.FV_TEXTLINE_CENTER (2)</li> <li>Constants.FV_TEXTLINE_MATH (3)</li> </ul>
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	<i>Color</i>	Denotes the spot color.

Property name	Data Type	Description
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	<i>Group</i>	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.



Property name	Data Type	Description
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FMOject	Denotes the next graphic object in the document.
NextGraphicInFrame	FMOject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FMOject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FMOject	Denotes the next selected graphic object in the document.

Property name	Data Type	Description
ObjectAttributes	<i>Strings</i>	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.

Property name	Data Type	Description
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Indicates the graphic object's internal unique number.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.
Direction	int	Set or get the direction of the document. Possible values: <ul style="list-style-type: none"> <li><i>Constants.FV_DIR_Inherit</i> - Inherit the direction of the parent</li> <li><i>Constants.FV_DIR_LTR</i> - <i>Left-to-right</i></li> <li><i>Constants.FV_DIR_RTL</i> - <i>Right-to-left</i></li> </ul>
ResolvedDirection	int	Get the inherited direction of the document. Possible values: <ul style="list-style-type: none"> <li><i>Constants.FV_DIR_LTR</i> - <i>Left-to-right</i></li> <li><i>Constants.FV_DIR_RTL</i> - <i>Right-to-left</i></li> </ul>

## TextLine methods

*Delete, GetProps, GetText, ObjectValid, SetProps.*

## TextLoc

Property name	Data Type	Description
obj	any	The object containing text.
offset	int	Characters from the start of the object.

## TextLoc methods

[TextLoc](#).

## TextRange

Property name	Data Type	Description
beg	<a href="#">TextLoc</a>	The beginning of the text range.
end	<a href="#">TextLoc</a>	The end of the text range.

## TextRange methods

[TextRange](#).

## TiApiClient

Property name	Data Type	Description
TiClientData	string	Data used by the client (for example, an SQL query).
TiClientName	string	The registered name of the client that created the inset.
TiClientSource	string	The name that appears as the source in the Text Inset Properties dialog box.
TiClientType	string	The name that appears as the source type in the Text Inset Properties dialog box.
TiIsUnresolved	int	1 if the inset is unresolved. A client should set this property to 0 if it is unable to resolve the inset.
ImportHint	string	Record identifying the filter used to import the text. The FrameMaker product uses this record to find the filter to use when updating the inset.

Property name	Data Type	Description
TiLocked	int	1 if the inset is locked. To change an inset's contents, you must unlock it. Always relock an inset after you have finished changing its contents.
Name	string	A name assigned to the inset by an FDK client. It is not automatically assigned by the FrameMaker product.
NextTiInDoc	FMOBJECT	The ID of the next text inset in the list of text insets in the document (Constants.FO_TiApiClient, Constants.FO_TiText, Constants.FO_TiTextTable, or Constants.FO_TiFlow ID).
TextRange	<i>TextRange</i>	The text range, in the document containing the text inset, occupied by the text inset.
TiAutomaticUpdate	int	1 if the inset is updated automatically. TiAutomaticUpdate has no effect if the document's DontUpdateTextInsets property is set to True.
TiFile	string	Path of the source file.
TiFileModDate	string	The modification date of the text inset's source file.
LastUpdate	int	Time when the inset was last updated, expressed in seconds since 1 January, 1970.
Unique	int	The text inset's UID.

## TiApiClient methods

[ConvertToText](#), [Delete](#), [DeletePropByName](#), [DeleteTextInsetContents](#), [GetProps](#), [GetText](#), [ObjectValid](#), [SetProps](#), [UpdateTextInset](#).

## TiFlow

Property name	Data Type	Description
TiFlowName	string	The name of the imported flow if TiMainFlow is False.
TiFlowPageSpace	int	The type of pages the imported flow is on: <ul style="list-style-type: none"> <li>Constants.FV_BODY_PAGE(0x00)</li> <li>Constants.FV_REFERENCE_PAGE(0x02)</li> </ul>
TiFormat	int	Source of the imported text's format: <ul style="list-style-type: none"> <li>Constants.FV_SourceDoc (0) - The text is formatted with formats from the source document.</li> <li>Constants.FV_EnclosingDoc (1) - The text is formatted with formats from the document into which it is imported.</li> <li>Constants.FV_PlainText (2) - The text is formatted as plain text.</li> </ul>
TiMainFlow	int	1 if the inset text is imported from the main flow of the source document.
TiRemovePageBreaks	int	1 if page breaks are removed from the text when TiFormat is set to Constants.FV_EnclosingDoc(1).
TiRemoveOverrides	int	1 if format overrides are removed from the text when TiFormat is set to Constants.FV_EnclosingDoc(1).
ImportHint	string	Record identifying the filter used to import the text. The FrameMaker product uses this record to find the filter to use when updating the inset.
TiLocked	int	1 if the inset is locked. To change an inset's contents, you must unlock it. Always relock an inset after you have finished changing its contents.

Property name	Data Type	Description
Name	string	A name assigned to the inset by an FDK client. It is not automatically assigned by the FrameMaker product.
NextTiInDoc	FMOBJECT	The ID of the next text inset in the list of text insets in the document (Constants.FO_TiApiClient (64), Constants.FO_TiText (62), Constants.FO_TiTextTable (63), or Constants.FO_TiFlow (61) ID).
TextRange	<i>TextRange</i>	The text range, in the document containing the text inset, occupied by the text inset.
TiAutomaticUpdate	int	1 if the inset is updated automatically. TiAutomaticUpdate has no effect if the document's DontUpdateTextInsets property is set to True.
TiFile	string	Path of the source file.
TiFileModDate	string	The modification date of the text inset's source file.
LastUpdate	int	Time when the inset was last updated, expressed in seconds since 1 January, 1970.
Unique	int	The text inset's UID.

## TiFlow methods

*ConvertToText, Delete, DeletePropByName, DeleteTextInsetContents, GetProps, GetText, ObjectValid, SetProps, UpdateTextInset.*

## TiText

Property name	Data Type	Description
TiEOLisEOP	int	1 if line ends in the imported text file are treated as paragraph ends.

Property name	Data Type	Description
TiTextEncoding	string	The <code>ImportHintString</code> for the text inset. If this property is not set to <code>Constants.FO_TiText (62)</code> or <code>Constants.FO_TiTextTable (63)</code> , the string is null.
ImportHint	string	Record identifying the filter used to import the text. The FrameMaker product uses this record to find the filter to use when updating the inset.
TiLocked	int	1 if the inset is locked. To change an inset's contents, you must unlock it. Always relock an inset after you have finished changing its contents.
Name	string	A name assigned to the inset by an ESTK client. It is not automatically assigned by the FrameMaker product.
NextTiInDoc	FMOBJECT	The ID of the next text inset in the list of text insets in the document ( <code>Constants.FO_TiApiClient(64)</code> , <code>Constants.FO_TiText (62)</code> , <code>Constants.FO_TiTextTable (63)</code> , or <code>Constants.FO_TiFlow (61) ID</code> ).
TextRange	<i>TextRange</i>	The text range, in the document containing the text inset, occupied by the text inset.
TiAutomaticUpdate	int	1 if the inset is updated automatically. <code>TiAutomaticUpdate</code> has no effect if the document's <code>DontUpdateTextInsets</code> property is set to <code>True</code> .
TiFile	string	Path of the source file.
TiFileModDate	string	The modification date of the text inset's source file.
LastUpdate	int	Time when the inset was last updated, expressed in seconds since 1 January, 1970.
Unique	int	The text inset's UID.



## TiText methods

*ConvertToText, Delete, DeletePropByName, DeleteTextInsetContents, GetProps, GetText, ObjectValid, SetProps, UpdateTextInset.*

## TiTextTable

Property name	Data Type	Description
TiByRows	int	1 if each paragraph in the imported text is converted to a row of table cells; False if each paragraph in the imported text is converted to a table cell.
TiTblTag	string	The table format tag of the imported table
TiHeadersEmpty	int	1 if the imported text is not used to fill the heading rows.
TiNumSeparators	int	If <code>TiSeparator</code> specifies a space, the number of spaces used as a separator to parse the text into table cells.
TiSeparator	string	If <code>TiByRows</code> is True, a string specifying a separator, such as a tab, used to parse the text into table cells.
TiNumCols	int	If <code>TiByRows</code> is False, the number of columns in the table.
TiNumHeaderRows	int	The number of heading rows in the table.
TiTextEncoding	string	The <code>ImportHintString</code> for the text inset. If this property is not set to <code>Constants.FO_TiText (62)</code> or <code>Constants.FO_TiTextTable (63)</code> , the string is null.
ImportHint	string	Record identifying the filter used to import the text. The FrameMaker product uses this record to find the filter to use when updating the inset.

Property name	Data Type	Description
TiLocked	int	1 if the inset is locked. To change an inset's contents, you must unlock it. Always relock an inset after you have finished changing its contents.
Name	string	A name assigned to the inset by an FDK client. It is not automatically assigned by the FrameMaker product.
NextTiInDoc	FMOBJECT	The ID of the next text inset in the list of text insets in the document (Constants.FO_TiApiClient (64), Constants.FO_TiText (62), Constants.FO_TiTextTable (63), or Constants.FO_TiFlow (61) ID).
TextRange	<i>TextRange</i>	The text range, in the document containing the text inset, occupied by the text inset.
TiAutomaticUpdate	int	1 if the inset is updated automatically. TiAutomaticUpdate has no effect if the document's DontUpdateTextInsets property is set to True.
TiFile	string	Path of the source file.
TiFileModDate	string	The modification date of the text inset's source file.
LastUpdate	int	Time when the inset was last updated, expressed in seconds since 1 January, 1970.
Unique	int	The text inset's UID.

## TiTextTable methods

*ConvertToText, Delete, DeletePropByName, DeleteTextInsetContents, GetProps, GetText, ObjectValid, SetProps, UpdateTextInset.*

## TypedVal

Property name	Data Type	Description
valType	int	The type of value. See the following table.
sval	string	String value.
ssval	<i>Strings</i>	Set of strings.
msval	<i>Metrics</i>	Set of metrics.
psval	<i>Points</i>	Set of points.
tsval	<i>Tabs</i>	Set of tabs.
tlval	<i>TextLoc</i>	Text location.
trval	<i>TextRange</i>	Text range.
adsval	<i>AttributeDefs</i>	Set of attribute definitions
asval	<i>Attribute</i>	A single attribute definition
csval	<i>ElementCatalogEntries</i>	Element Catalog values.
isval	<i>Ints</i>	Set of integers.
osval	Objects	
uisval	<i>UInts</i>	Set of unsigned integers.
valsval	<i>TypedVals</i>	Set of TypedVal entries.
obj	any	Set of objects.
ival	int	An integer.

## TypedVals

An Array of *TypedVal* objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## TypedVals methods

[concat](#), [pop](#), [push](#).

## UBytes

An Array of objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## UBytes methods

[concat](#), [pop](#), [push](#).

## UInts

An Array of objects with integer indexing and a *length* property.

Property name	Data Type	Description
length	number	The length of the array

## UInts methods

[concat](#), [pop](#), [push](#).

## UnanchoredFrame

Property name	Data Type	Description
Name	string	The name of the frame.
PageFramePage	FObject	The page that the unanchored frame belongs to, if the unanchored frame is a page frame. The page can be one of:FO_HiddenPage, FO_BodyPage, FO_MasterPage, or FO_RefPage ID.

Property name	Data Type	Description
FirstGraphicInFrame	FObject	First object in the frame (backmost object).
LastGraphicInFrame	FObject	Last object in the frame (frontmost object).
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>• <code>Constants.FV_ARROW_STICK (0x1)</code> <i>Constants.FV_ARROW_HOLLOW (0x2)</i> <i>Constants.FV_ARROW_FILLED (0x3)</i></li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	<i>Color</i>	Denotes the spot color.

Property name	Data Type	Description
Dash	<i>Metrics</i>	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	<i>Group</i>	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.

Property name	Data Type	Description
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is -216 inches to 216 inches. If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FObject	Denotes the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Denotes the next selected graphic object in the document.

Property name	Data Type	Description
ObjectAttributes	<i>Strings</i>	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TR_NONE (0x01)</li> <li>Constants.FV_TR_CONTOUR (0x02)</li> <li>Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.



Property name	Data Type	Description
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Indicates the graphic object's internal unique number.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## UnanchoredFrame methods

*Delete, GetProps, ObjectValid, SetProps.*

## Var

Property name	Data Type	Description
Element	<i>Element</i>	The element associated with the variable, if the variable is in a FrameMaker document.
Locked	int	1 if the variable is included in a text inset that gets its formatting from the source document. The variable is not affected by global formatting performed on the document.
NextVarInDoc	<i>Var</i>	Next variable instance (FO_Var ID) in the document.
TextRange	<i>TextRange</i>	The text range the variable instance encompasses.
VarFmt	<i>VarFmt</i>	The variable instance's format (FO_VarFmt ID).
Unique	int	The variable's unique ID.
UserString	string	A string to which clients can store private data.

## Var methods

[Delete](#), [GetProps](#), [GetText](#), [ObjectValid](#), [SetProps](#).

## VarFmt

Property name	Data Type	Description
Fmt	string	The variable format definition; the building blocks and text strings used to create a variable instance with the variable format.
Name	string	The variable format's name.
NextVarFmtInDoc	<a href="#">VarFmt</a>	Next variable format (FO_VarFmt ID) in the document's list of variable formats.

Property name	Data Type	Description
SystemVar	int	<p>The variable format's type. The following types specify system variable formats:</p> <ul style="list-style-type: none"> <li>           Constants.FV_VAR_CURRENT_PAGE_NUM (1)  <i>Constants.FV_VAR_PAGE_COUNT</i> (2)  <i>Constants.FV_VAR_CURRENT_DATE_LONG</i> (3)  <i>Constants.FV_VAR_CURRENT_DATE_SHORT</i> (4)  <i>Constants.FV_VAR_MODIFICATION_DATE_LONG</i> (5)  <i>Constants.FV_VAR_MODIFICATION_DATE_SHORT</i> (6)  <i>Constants.FV_VAR_CREATION_DATE_LONG</i> (7)  <i>Constants.FV_VAR_CREATION_DATE_SHORT</i> (8)  <i>Constants.FV_VAR_FILE_NAME_LONG</i> (9)  <i>Constants.FV_VAR_FILE_NAME_SHORT</i> (10)  <i>Constants.FV_VAR_HEADER_FOOTER_1</i> (11) through <i>Constants.FV_VAR_HEADER_FOOTER_4</i> (14)  <i>Constants.FV_VAR_TABLE_CONTINUATION</i> (15)  <i>Constants.FV_VAR_TABLE_SHEET</i> (16)  <i>Constants.FV_VAR_HEADER_FOOTER_5</i> (19) through <i>Constants.FV_VAR_HEADER_FOOTER_12</i> (26)         </li> </ul> <p>The following type specifies user defined formats:  <i>Constants.FV_VAR_USER_VARIABLE</i> (0) - A user defined variable format.</p>

## VarFmt methods

*Delete, GetProps, ObjectValid, SetProps.*

## XRef

Property name	Data Type	Description
Element	<i>Element</i>	The associated element, if the cross reference is in a FrameMaker document.
Locked	int	1 if the cross-reference is part of a text inset that retains formatting information from the source document. The cross-reference is not affected by global formatting performed on the document.
NextXRefInDoc	<i>XRef</i>	Next cross-reference instance in document (FO_XRef ID).
TextRange	<i>TextRange</i>	Text range that the cross-reference instance encompasses.
Unique	int	The cross-reference's unique ID.
UserString	string	A string to which clients can store private data.
XRefFmt	<i>XRefFmt</i>	ID of the cross-reference's format (FO_XrefFmt ID).
XRefFile	string	The filename of the file containing the cross-reference source. If the cross-reference source is in the same document as the cross reference, the filename is an empty string ("").
XRefsUnresolved	int	1 if FrameMaker was unable to resolve the cross-reference the last time it updated crossreferences. <b>NOTE:</b> <i>This property is set only when FrameMaker updates cross-references. Changes to the document, in and of themselves, do not affect this property.</i>

Property name	Data Type	Description
XRefSrcText	string	If XRefSrcIsElem is False, the XRefSrcTex property is a string specifying <code>UID:pgf_tag:text</code> , where UID represents the unique ID for the paragraph, <code>pgf_tag</code> is the name of the paragraph format, and <code>text</code> is the text content of the paragraph. If XRefSrcIsElem is True, the XRefSrcTex property is a string specifying <code>UID:src_name:text</code> , where UID is the value of the ID attribute of the source element, <code>name</code> is the element tag, and <code>text</code> is text content of the source element.
XRefSrcIsElem	int	1 if the cross-reference source is a FrameMaker structural element.
XRefSrcElemNonUniqueId	string	If XRefSrcIsElem is True, this is a string specifying the value of <code>@id</code> of the source element inside the element identified by the Unique ID in XRefSrcText. This is required in cases where the Id of the source element is not a Unique ID.
XRefAltText	string	Alternate text for display as the cross-reference link text. This is given preference, if present, over the earlier behavior of getting text from the source.
XRefClientName	string	The registered name of the client that created the cross-reference.
XRefClientType	string	The identifier for specifying the source type for the cross-reference.

## XRef methods

*Delete, GetProps, GetText, ObjectValid, SetProps.*

## XRefFmt

Property name	Data Type	Description
Fmt	string	The cross-reference format (a string that specifies text and building blocks).
Name	string	The cross-reference format's name.
NextXRefFmtInDoc	<i>XRefFmt</i>	Denotes the next cross-reference format (FO_XRefFmt ID).

### XRefFmt methods

*Delete, GetProps, ObjectValid, SetProps.*

## app

Property name	Data Type	Description
ActiveBook	<i>Book</i>	The book with input focus (FO_Book ID).
ActiveDoc	<i>Doc</i>	The document with input focus (FO_Doc ID).
ActiveView	string	Sets the current view. The view can be one of: WYSIWYG View Author View XML View
ApplyFormatRules	int	1 if element reformatting is enabled.
AddMarkerTypeToStandardMarkers	string	The name of a marker type to add to the standard list of marker types. Use <code>SetString()</code> to set a marker type name to this property of the <code>Constants.FV_SessionId(0)</code> .
AutoBackup	int	1 if Automatic Backup is enabled.
AutoSave	int	1 if Automatic Save is enabled.

Property name	Data Type	Description
AutoSaveSeconds	int	Time between automatic saves in seconds (60 seconds to 10800 seconds).
Displaying	int	False if screen refresh is completely turned off.
DisplayName	string	The name of the display on which the FrameMaker product session is running.
DefaultKeyCatalog	Doc	Default Key Catalog for the current workflow. Workflow values: KeyCatalogWorkflowAuthoring KeyCatalogWorkflowPublishing KeyCatalogWorkflowSearchIng
FirstKeyCatalogInSession	Doc	First Key Catalog in the session.
KeyCatalogWorkflow	int	Current workflow related to Key Catalogs.
IsTempOpenSave	int	Gets whether temporary open/save is in progress. Temporary open/save happens during view switching operations.
ExportFilters	string	List of export filters available in the current session.
FirstCommandInSession	<i>Command</i>	First command in the list of commands in the session (FO_Command ID).
FirstMenuItemInSession	FMOBJECT	First menu item or menu in the list of menus, menu items, and menu item separators in the session (Constants.FO_Command (56), Constants.FO_Menu (55), Constants.FO_MenuItemSeparator (57) ID).
FirstOpenBook	<i>Book</i>	First open book (FO_Book ID) in session.
FirstOpenDoc	<i>Doc</i>	First open document (FO_DocID) in session.
BinDir	string	Directory path of \$FMHOME/bin.

Property name	Data Type	Description
CurrentDir	string	Name of the directory from which the FrameMaker product was started
CurrentMenuSet	int	Type of menu set. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_MENU_QUICK (1)</li> <li>Constants.FV_MENU_COMPLETE (2)</li> <li>Constants.FV_MENU_CUSTOM (3)</li> </ul>
HelpDir	string	Path of the FrameMaker product help directory
HomeDir	string	Path of \$FMHOME
UserSettingsDir	string	Name of the FrameMaker app directory. As defined by: %appdata%\Adobe\FrameMaker\ version number> Returns the HomeDir if the app directory is not found.
importFilters	string	List of import filters available in the current session.
InitDir	string	Directory path of \$FMHOME/fm_init
FontAngleNames	<i>Strings</i>	List of font angles available in the current session.



Property name	Data Type	Description
FontFamilyAttributes	<i>Ints</i>	<p>An array of flags that indicate attributes for each font family listed by FontFamilyNames. This array of integers is indexed the same as the list of font family names, and corresponds directly to that list. Each Int is a packed field; the high order 16 bits indicate a surrogate font, and the low order bits indicate attributes for the font family. The flags, their mask values, and their meaning follow:</p> <ul style="list-style-type: none"> <li>Constants.FV_FAMILY_VISIBLE (0x00000001) - Family is visible in menu.</li> </ul> <p><i>Constants.FV_FAMILY_SELECTABLE (0x00000002) - Family can be selected in menu.</i></p> <p><i>Constants.FV_FAMILY_MAPPED (0x00000004) - Family is always mapped to another family.</i></p> <p><i>Constants.FV_FAMILY_SURROGATE (0xFFFF0000): The family mapped to, if Constants.FV_FAMILY_MAPPED (0x00000004) is True.</i></p>
FontFamilyNames	<i>Strings</i>	List of font family names available in the current session. This list does not include combined fonts.
FontVariationNames	<i>Strings</i>	List of font variations available in the current session.
FontWeightNames	<i>Strings</i>	List of font weights available in the current session.
Gravity	int	1 if Gravity is turned on for the session.
GreekSize	int	Size at which to greek text.
HostName	string	Name of the host computer.

Property name	Data Type	Description
IconBarOn	<code>int</code>	1 if the four icons that appear on the upper-right side of the document window are on. Changing this property affects only the documents that are opened subsequently; it does not affect documents that are already open.
IsIconified	<code>int</code>	1 if the FrameMaker product window is iconified.
IsInFront	<code>int</code>	1 if the FrameMaker product window is in front of other application windows. You can use this property to bring the FrameMaker product to the front or back.
IsOnScreen	<code>int</code>	1 if the FrameMaker product window is visible on the screen.
Label	<code>string</code>	The title in the FrameMaker product window title bar.

Property name	Data Type	Description
Language	int	Product language. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li><i>Constants.FV_LANG_ENGLISH (0x01)</i></li> <li><i>Constants.FV_LANG_BRITISH (0x02)</i></li> <li><i>Constants.FV_LANG_GERMAN (0x03)</i></li> <li><i>Constants.FV_LANG_FRENCH (0x05)</i></li> <li><i>Constants.FV_LANG_SPANISH (0x07)</i></li> <li><i>Constants.FV_LANG_ITALIAN (0x09)</i></li> <li><i>Constants.FV_LANG_SWEDISH (0x11)</i></li> <li><i>Constants.FV_LANG_JAPANESE (0x12)</i></li> <li><i>Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</i></li> <li><i>Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</i></li> <li><i>Constants.FV_LANG_KOREAN (0x15)</i></li> </ul>
MarkerNames	<i>Strings</i>	List of standard marker types for the current session. For versions prior to 5.5, this property returned the list of all marker types for the current session. In version 5.5, marker types are assigned to the document; use the <code>MarkerTypeNames</code> property of <code>Constants.FO_Doc (4)</code> to get the full list of marker types.
OpenDir	string	Directory in which FrameMaker is installed.
Path	string	Path to search to start the FrameMaker product.

Property name	Data Type	Description
Platform	string	Name of the platform on which the current session is running.
ProductIsDemo	bool	1 if the current session is for a demo version of FrameMaker.
ProductIsStructured	bool	1 if FrameMaker is running in structured mode for the current session.
ProductName	string	The FrameMaker product name. The names for FrameMaker+SGML indicate FrameMaker running under the structured product interface. FrameViewer is retained for backward compatibility. Can be one of: <ul style="list-style-type: none"> <li>FrameMaker</li> <li><i>FrameMaker+SGML</i></li> <li><i>FrameViewer</i></li> <li><i>DemoMaker</i></li> <li><i>DemoMaker+SGML</i></li> </ul>
ProductFamily	string	The FrameMaker product family name.: <ul style="list-style-type: none"> <li>FrameMaker</li> <li><i>FrameMaker XML Author</i></li> </ul>
ViewQuickAccessBar	int	1 if the QuickAccess bar is visible.
ViewFormattingBar	int	1 if the formatting bar is visible.
Reformatting	int	1 if reformatting is enabled.
RememberMissingFontNames	int	1 if Remember Missing Font Names is activated.
ScreenHeight	int	Height of the FrameMaker product window in pixels.
ScreenWidth	int	Width of the FrameMaker product window in pixels.
ScreenX	int	The offset of the FrameMaker product window in pixels from the left side of the screen. If you set a value that would result in the product window being off the screen, that value is ignored and the old value is retained.

Property name	Data Type	Description
ScreenY	int	The offset of the FrameMaker product window in pixels from the top of the screen. If you set a value that would result in the product window being off the screen, that value is ignored and the old value is retained.
SgmlDir	string	Directory path of \$FMHOME/structure/sgml.
Snap	int	1 if Snap is turned on for the session.
StructureDir	string	Directory path of \$FMHOME/structure.
TmpDir	string	Path of the temporary directory for internal FrameMaker product processes; the directory specified by the DOS \$TEMP environment variable.
UserLogin	string	User login name.
UserName	string	User name.
Validating	int	1 if validation is enabled (FrameMaker only).
VersionMajor	int	Frame version number (before the decimal).
VersionMinor	int	Frame version number (after the decimal).
WindowSystem	string	Name of window system that the FrameMaker product is running under.
XmlDir	string	Directory path of \$FMHOME/structure/xml.
DoNotGenerateErrorLog	int	When 1, the FrameMaker error log is not generated.
NoFlashInPDF	bool	When True, the Flash content in the PDF becomes static. When False, Flash objects get embedded in the PDF and are playable. Default is False.

Property name	Data Type	Description
No3DInPDF	bool	When <code>True</code> , the 3D objects in the generated PDF become 2-dimensional. When <code>False</code> , 3D objects get embedded in the PDF and do not become 2 dimensional. Default is <code>False</code> .
DontShowWelcomeScreen	bool	<code>True</code> when FrameMaker is set not to show the Welcome screen.
EnableAutoSpellCheck	bool	When <code>True</code> , Auto Spell Check is enabled.
AllowNewFileURL	bool	When <code>True</code> , allows usage of <code>file:/</code> and <code>file:</code> URLs for referring to a file.
CharacterReferencesOnXMLExport	bool	When <code>True</code> , if there are leading/trailing whitespaces (spaces and tabs) in elements, save the whitespaces as character references to enable round-tripping. Else set to <code>False</code> . Default: <code>False</code>
DoNotExportInvalidXML	bool	When <code>True</code> , FrameMaker does not save the XML on disk if there are validation errors during XML export.
DoNotExportInvalidXML	bool	When <code>True</code> , FrameMaker does not save the XML on disk if there are validation errors during XML export.
SuppressXMLParserWarnings	bool	When <code>True</code> , the warning messages about duplicate declarations in DTD/Schema do not appear in the FrameMaker log.
RemoveExtraWhiteSpacesOnXMLImport	bool	When <code>True</code> , the extra while spaces are removed while importing XML.
NoMultiMediaInPDF	bool	When <code>True</code> , the Rich Media (audio and video) objects in the PDF become static. When <code>False</code> , Rich Media objects get embedded in the PDF and are playable. Default is <code>False</code> .

Property name	Data Type	Description
AutoScaleImageAlongWidth	bool	When <code>True</code> , the images (upon insertion) get scaled along the width of the parent text frame.

## app methods

*Close, GetNamedBook, GetNamedCommand, GetNamedMenu, GetNamedMenuItemSeparator, GetNamedObject, GetProps, NewNamedBook, NewNamedCommand, NewNamedMenu, NewNamedMenuItemSeparator, NewNamedObject, ObjectValid, SetProps.*

# Function Summary

## AFrame

### Delete

#### Description

The `Delete()` method is used to delete an object from a document. When you delete an object that contains another object, all child objects are also deleted with the parent object. For example, if you delete a frame, all objects within the frame are also deleted.

The `Delete()` method does not take any arguments.

The method returns `FE_SUCCESS` on success, else returns one of the following values in `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadDelete (-17)	Specified object could not be deleted.
Constants.FE_BadOperation (-27)	Function call specified an illegal operation.
Constants.FE_BadParameter (-43)	Function call specified an invalid parameter.
Constants.FE_NotMenu (72)	Object is a menu item but the document to delete does not belong to the menu.

#### Returns

`int`

#### Syntax

`Delete()`



## GetProps

### Description

The `GetProps()` method retrieves the complete property list for a specified object.

The `GetProps()` method takes no arguments.

The method returns a `PropVals` object that contains an array of property-value pairs, on success.

If the `GetProps()` method fails, it sets the `len` field of the returned structure to 0 and assigns one of the following values to `FA_erno`.

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the specified operation.

### Returns

*PropVals*

### Syntax

`GetProps()`

## SetProps

### Description

The `SetProps()` method sets the properties of a specified object.

The method takes one argument - An array containing the values of the object to be set.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadName (-33)	Specified name is illegal.
Constants.FE_BadNewFrame (-14)	The method cannot move the specified object to this frame.

Error	Reason
Constants.FE_BadNewGroup (-15)	The method cannot move the specified object to this graphic object group (FO_Group).
Constants.FE_BadNewSibling (-16)	Object cannot be made a sibling of the specified object.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadPropNum (-4)	Specified property number is invalid.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_CantSmooth (-38)	Object cannot be smoothed.
Constants.FE_DupName (-32)	Property cannot be set to this name because it is already used by another object.
Constants.FE_GenRuleAmbiguous (-55)	General rule in structured document is ambiguous.
Constants.FE_GenRuleConnectorExpected (-57)	General rule in structured document is missing a connector.
Constants.FE_GenRuleItemExpected (-51)	General rule in structured document is missing a rule item.
Constants.FE_GenRuleLeftBracketExpected (-53)	General rule in structured document is missing a left bracket.
Constants.FE_GenRuleMixedConnectors (-52)	General rule in structured document has mixed connectors.
Constants.FE_GenRuleRightBracketExpected (-54)	General rule in structured document is missing a right bracket.
Constants.FE_GenRuleSyntaxError (-56)	General rule in structured document has a syntax error.
Constants.FE_GroupSelect (-9)	The method cannot select or deselect an object in the specified group.
Constants.FE_HiddenPage (-40)	The specified value must be a hidden page (FO_HiddenPage).
Constants.FE_InvContextSpec ()	The method encountered an invalid context specification in a FrameMaker document.
Constants.FE_NotBookComponent (-26)	The specified value must be a book component (FO_BookComponent).
Constants.FE_NotFrame (-12)	The specified value must be a frame.

Error	Reason
Constants.FE_NotGraphic (-11)	The specified value must be a graphic object.
Constants.FE_NotGroup (-13)	The specified value must be a graphic object group (FO_Group).
Constants.FE_NotTextFrame (-39)	The specified value must be a text column (FO_TextFrame).
Constants.FE_NotTextObject (-41)	The specified object must be a text object, such as a paragraph (FO_Pgf) or a flow (FO_Flow).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.
Constants.FE_OutOfRange (-7)	Specified property value is out of the legal range for the specified property.
Constants.FE_PageFrame (-37)	The specified value must be a page frame object (FO_UnanchoredFrame).
Constants.FE_ReadOnly (-6)	Property is read-only.
Constants.FE_WithinFrame (-10)	The specified object must first be moved to a different frame.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support this operation.

**Returns**

Void

**Syntax**

SetProps (setVal)

**Parameters**

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list

**ObjectValid****Description**

Indicates whether an object is valid or not.

The method does not take any arguments.

The method returns 1 if the object is valid, else returns 0.

**Returns**

`int`

**Syntax**

`ObjectValid()`

## Arc

### Delete

**Description**

Deletes the Arc object.

See [Delete](#) under the AFrame class for more information.

**Returns**

`int`

**Syntax**

`Delete()`

### GetProps

**Description**

The `GetProps()` method retrieves the properties of the Arc object.

See [SetProps](#) under the AFrame class for more information.

**Returns**

*PropVals*

**Syntax**

`GetProps()`

## SetProps

### Description

The `SetProps()` method sets the properties of the Arc object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the Arc object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## AttrCondExpr

### ApplyAttributeExpression

#### Description

`ApplyAttributeExpression()` is the method to apply the attribute expression to the document.

The method returns `FE_Success` on success.

### Returns

`int`

### Syntax

```
ApplyAttributeExpression()
```

## GetProps

### Description

The `GetProps()` method gets the properties of the specified attribute expression.

See [GetProps](#) under the `AFrame` class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the specified attribute expression.

See [SetProps](#) under the `AFrame` class for more information.

### Returns

`Void`

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>setVal</code>	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the attribute expression is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Attribute

### Attribute

### Description

Describes a single attribute.

### Returns

[Attribute](#)

### Syntax

```
Attribute(name, values, valflags, allow)
```

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	Attribute name.
values	<i>Strings</i>	No	Attribute values.
valflags	uint	No	Validation error flags.
allow	uint	No	Allow error as special case.

# AttributeDef

## AttributeDef

### Description

Describes a single attribute function.

### Returns

*AttributeDef*

### Syntax

```
AttributeDef(name, required, flags, attrType, choices, defValues,
rangeMin, rangeMax)
```

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	Attribute name.
required	int	No	1 if the attribute is required; 0 otherwise.
flags	uint	No	One of read-only, hidden or neither.
attrType	int	No	The attribute type.
choices	<i>Strings</i>	No	The list of choices if attrType is set to Constants.FV_AT_CHOICES (2)
defValues	<i>Strings</i>	No	The default value, if attribute is not mandatory.
rangeMin	string	No	The minimum allowed value (if any)
rangeMax	string	No	The maximum allowed value (if any)

The `flags` parameter indicates whether the attribute is read-only, hidden or neither.

Flag	Meaning
Constants.FV_AF_READ_ONLY(0x0001)	The attribute value is read-only.
Constants.FV_AF_HIDDEN(0x0002)	The attribute value is hidden.



Flag	Meaning
NULL	The attribute value is neither read-only nor hidden.

The `attrType` parameter can have one of the following values:

Value	Meaning
Constants.FV_AT_STRING(0)	Any arbitrary text string.
Constants.FV_AT_STRINGS(1)	One or more arbitrary text strings.
Constants.FV_AT_CHOICES(2)	A value from a list of choices.
Constants.FV_AT_INTEGER(3)	A signed whole number (optionally restricted to a range of values).
Constants.FV_AT_INTEGERS(4)	One or more integers (optionally restricted to a range of values).
Constants.FV_AT_REAL(5)	A real number (optionally restricted to a range of values).
Constants.FV_AT_REALS(6)	One or more real numbers (optionally restricted to a range of values).
Constants.FV_AT_UNIQUE_ID(7)	A string that uniquely identifies the element.
Constants.FV_AT_UNIQUE_IDREF(8)	A reference to a UniqueID attribute.
Constants.FV_AT_UNIQUE_IDREFS(9)	One or more references to a UniqueID attribute.

## AttributeDefs

An Array of [AttributeDef](#) objects with integer indexing and a *length* property.

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

[AttributeDefs](#)

**Syntax**

```
concat (value)
```

**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>AttributeDefs</i>	No	Can also be arrays.

**pop****Description**

Removes the last element from the array.

**Returns**

*AttributeDef*

**Syntax**

```
pop ()
```

**push****Description**

Returns the new length of the array.

**Returns**

number

**Syntax**

```
push (value)
```

**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>AttributeDef</i>	No	Value to push into the array.

## Attributes

An Array of *Attribute* objects with integer indexing and a *length* property.

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

*Attributes*

#### Syntax

`concat (value)`

#### Parameters

Parameter name	Data Type	Optional	Description
value	<i>Attributes</i>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

#### Returns

*Attribute*

#### Syntax

`pop ()`

### push

#### Description

Returns the new length of the array.

**Returns**

number

**Syntax**`push(value)`**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>Attribute</i>	No	Value to push into the array.

**AttributesEx**

An Array of *AttributeEx* objects with integer indexing and a *length* property.

**concat****Description**

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array as the result of concatenation, with the given values to the end of the original array.

**Returns***AttributesEx***Syntax**`concat(value)`**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>AttributesEx</i>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

*AttributeEx*

### Syntax

```
pop()
```

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

```
push(value)
```

### Parameters

Parameter name	Data Type	Optional	Description
value	<i>AttributeEx</i>	No	Value to push into the array.

## BodyPage

### Delete

#### Description

Deletes the specified Body Page.

See [Delete](#) under the AFrame class for more information.

**Returns**`int`**Syntax**`Delete()`**GetProps****Description**

The `GetProps()` method retrieves the properties of the specified Body Page.

See [GetProps](#) under the `AFrame` class for more information.

**Returns***PropVals***Syntax**`GetProps()`**SetProps****Description**

The `SetProps()` method sets the properties of the specified Body Page.

See [SetProps](#) under the `AFrame` class for more information.

**Returns**`Void`**Syntax**`SetProps(setVal)`**Parameters**

Parameter name	Data Type	Optional	Description
<code>setVal</code>	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the Body Page is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## ApplyPageLayout

### Description

The `ApplyPageLayout()` method applies the layout of one page to another page.

The method returns `FE_Success` on success, and sets `FA_errno` with one of the following values on error:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation or <code>fmbatch</code> is running.
Constants.FE_BadOperation (-27)	Parameters specified an invalid operation.
Constants.FE_BadParameter (-43)	The specified parameter has an invalid value.
Constants.FE_SystemError (-42)	System Error.

### Returns

int

### Syntax

```
ApplyPageLayout (srcPage)
```

## Parameters

Parameter name	Data Type	Optional	Description
srcPage	PageObject	No	The page to which the layout is to be applied.

## Book

### Compare

#### Description

The `Compare()` method compares the differences between two versions of files and stores the result in a `CompareRet` data object.

You can OR the following values into the flags parameter:

Flag	Meaning
<code>Constants.FF_CMP_SUMMARY_ONLY (0x01)</code>	Generate a summary document, but not a composite document.
<code>Constants.FF_CMP_CHANGE_BARS (0x02)</code>	Turn on change bars in the composite document.
<code>Constants.FF_CMP_HYPERLINKS (0x04)</code>	Put hypertext links in the summary document.
<code>Constants.FF_CMP_SUMKIT (0x08)</code>	Open the summary document.
<code>Constants.FF_CMP_COMPKIT (0x10)</code>	Open the composite document.

The returned `CompareRet` object contains the pointers to the summary and composite results documents.

The method returns the comparison results in the `CompareRet` object on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.



Error	Reason
Constants.FE_BadCompare (-35)	Older and newer documents are not the same types of files.
Constants.FE_CompareTypes (-34)	One of the files is not a FrameMaker document or book or one file is a book and the other is a document.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the operation.

## Returns

*CompareRet*

## Syntax

`Compare(newerDoc, flags, insertCondTag, deleteCondTag, replaceText, compareThreshold)`

## Parameters

Parameter name	Data Type	Optional	Description
<code>newerDoc</code>	<i>Doc</i>	No	The newer version of the document or book.
<code>flags</code>	<code>int</code>	No	Bit flags that specify how to generate the summary and composite documents. Specify 0 for the default flags.
<code>insertCondTag</code>	<code>string</code>	No	Indicates the condition tag to apply to insertions shown in the composite document. To specify a blank insert condition tag, enter the value <code>null</code> .
<code>deleteCondTag</code>	<code>string</code>	No	Denotes the condition tag to apply to deletions shown in the composite document. To specify a blank delete condition tag, enter the value <code>null</code> .
<code>replaceText</code>	<code>string</code>	No	Text to appear in place of the deleted text. To not specify any replacement text, enter the value <code>null</code> .

Parameter name	Data Type	Optional	Description
compareThreshold	int	No	Threshold expresses the percentage of words that can change before paragraphs are considered not equal. If two paragraphs are equal, word differences between them are shown within a paragraph in the composite document. If the compareThreshold property is set to 0, the Compare() method uses the default value of 75. If the compareThreshold property is set to any other value, the Compare() method uses the setting in the FrameMaker configuration files.

## NewBookComponentInHierarchy

### Description

The `NewBookComponentInHierarchy()` method inserts a book component at a specified position in a FrameMaker structured book, and returns the component that was newly added, on success.

The method returns the element that was added, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadBookId (-66)	Invalid book.
Constants.FE_BadCompPath (-68)	Component name specified for compName is invalid.
Constants.FE_BadNew (-23)	The object cannot be created.
Constants.FE_BookUnStructured (-67)	The specified book is unstructured.

### Returns

*Element*

### Syntax

```
NewBookComponentInHierarchy(compName, elemLoc)
```

## Parameters

Parameter name	Data Type	Optional	Description
compName	string	No	The name of the component to add.
elemLoc	<i>Element Loc</i>	No	The position at which to add the new component.

## SilentPrintDoc

### Description

The `SilentPrintDoc()` method prints a document or a book using the default print settings. Default print settings are the settings that appear in the Print dialog box when the user attempts to print a document.

`SilentPrintDoc()` initializes the print page size and printer name if they do not have values.

On success, the method returns `FE_Success`. On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_SystemError</code> (-42)	Could not open or close the printer file.
<code>Constants.FE_BadParameter</code> (-43)	Parameter has an invalid value.

### Returns

`int`

### Syntax

```
SilentPrintDoc()
```

## SimpleGenerate

### Description

The `SimpleGenerate()` method generates files for a book.

The method performs the same operation as choosing `Update Book` from the `book Edit` menu.

You must set up the book and its generated files before calling `SimpleGenerate()`.

The method returns `FE_Success` on success, and assigns `FA_errno` with one of the following values on failure:

Error	Reason
<code>Constants.FE_BadOperation (-27)</code>	The book is not self-consistent (book generates data in one file that is source data for another generated file, or page count continually changes for this operation); there is a duplicate file in the book; all files in the book are generated files.
<code>Constants.FE_BadParameter (-43)</code>	Not a valid book.
<code>Constants.FE_SystemError (-42)</code>	Could not allocate memory, or could not open or save one of the files in the book.

### Returns

`int`

### Syntax

```
SimpleGenerate(interactive, makeVisible)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>interactive</code>	<code>int</code>	No	Set the interactive property to 1 to display messages and warnings; set the property to 0 to suppress display of messages and warnings.
<code>makeVisible</code>	<code>int</code>	No	Specifies whether to display the generated files. Set to 1 to display the files, or to 0 to suppress displaying the files.

## SimpleImportFormats

### Description

The `SimpleImportFormats()` method imports formats from a document to a document or a book.

**NOTE:** If you import formats to a book, `SimpleImportFormats()` imports formats to each book component for which the `ImportFmtInclude` property is set to `True`.

You can OR the following values into the `formatFlags` parameter to specify the formats to import.

Flag	What to Import
Constants.FF_UFF_COLOR (0x0100)	Colors
Constants.FF_UFF_COMBINED_FONTS (0x0800)	Combined fonts
Constants.FF_UFF_COND (0x0010)	Conditions
Constants.FF_UFF_DOCUMENT_PROPS (0x0400)	Document properties
Constants.FF_UFF_FONT (0x0002)	Character Catalog formats
Constants.FF_UFF_MATH (0x0200)	Equation settings
Constants.FF_UFF_PAGE (0x0004)	Page layouts
Constants.FF_UFF_PGF (0x0001)	Paragraph Catalog formats
Constants.FF_UFF_REFPAGE (0x0020)	Reference pages
Constants.FF_UFF_REMOVE_EXCEPTIONS (0x8000)	Remove exception formats from target documents
Constants.FF_UFF_REMOVE_PAGE_BREAKS (0x4000)	Remove all forced page breaks from target documents
Constants.FF_UFF_TABLE (0x0008)	Table Catalog formats
Constants.FF_UFF_VAR (0x0040)	Variable formats
Constants.FF_UFF_XREF (0x0080)	Cross-reference formats

The method returns `FE_Success` on success. If the method fails, it sets `FA_errno` with one of the following values:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support books.
Constants.FE_BadDocId (-2)	Invalid book or document.
Constants.FE_Canceled (-44)	User canceled the operation.
Constants.FE_FailedState (-45)	FrameMaker failed to open one or more of the book's document files during the import operation.

**Returns**

int

**Syntax**

SimpleImportFormats(fromDoc, formatFlags)

**Parameters**

Parameter name	Data Type	Optional	Description
fromDoc	<i>Doc</i>	No	The document from which the formats are to be imported.
formatFlags	int	No	Bit field specifying the formats to be imported. Specify 0 for the default flags.

**SimpleImportElementDefs****Description**

The `SimpleImportElementDefs()` method imports element definitions and the format change list catalog from an EDD or FrameMaker document or book to another FrameMaker document or book.

You can OR the following flags into the `importFlags` parameter:

Flag	Meaning
Constants.FF_IED_REMOVE_OVERRIDES (0x0001)	Clear format overrides.
Constants.FF_IED_REMOVE_BOOK_INFO (0x0002)	Clears formatting inherited from a parent book, if the element definitions are being imported from a document.
Constants.FF_IED_DO_NOT_IMPORT_EDD (0x0004)	If the source document is an EDD, setting this value does NOT treat it as an EDD; instead the method just imports its element catalog.
Constants.FF_IED_NO_NOTIFY (0x0008)	Do not issue the <code>Constants.FA_Note_PreImportElemDefs (64)</code> or <code>Constants.FA_Note_PostImportElemDefs (65)</code> notifications.

The method returns `FE_Success` on success. If the method fails, it assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current product version is not Structured FrameMaker.
Constants.FE_BadDocId (-2)	Invalid book or document.

**Returns**

int

**Syntax**

```
SimpleImportElementDefs (fromDocOrBook, importFlags)
```

**Parameters**

Parameter name	Data Type	Optional	Description
fromDocOrBook	Doc Book	No	The document or book from which to import element definitions.
importFlags	int	No	Specifies how to import formats.

**GetUniqueObject****Description**

The `GetUniqueObject()` method retrieves the object from its persistent unique identifier (UID). The FrameMaker product assigns a UID to each object in a document or book that is not identified by a name. The UID, unlike the object's ID, does not change from one session to another.

The `GetUniqueObject()` method returns the object on success. On failure, it sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document or book.
Constants.FE_TypeUnNamed (-19)	Objects of the specified type are not identified by UIDs.

Error	Reason
Constants.FE_NameNotFound (-20)	Object with specified UID could not be found.

**Returns**

Object

**Syntax**

GetUniqueObject(objType, unique)

**Parameters**

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object (for example, FO_Pgf).
unique	int	No	The unique ID of the object.

**Close****Description**

Closes a document, book, dialog box, or Frame session.

The `flags` parameter specifies whether to abort or to close open documents or books if they have unsaved changes. Set the `Constants.FF_CLOSE_MODIFIED` flag to `True (1)` to close open documents and books regardless of their state.

The method sets `FE_Success` on success. On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_DocModified (-8)	The document was modified and the flags parameter was set to 0.

**Returns**

int

**Syntax**

Close(flags)



## Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	The Close() method closes the document or book. <i>NOTE: Flags specify whether to abort or to close open documents or books if they have unsaved changes. Set the FF_CLOSE_MODIFIED flag to True (1) to close open documents and books regardless of their state.</i>

## SimpleSave

### Description

The SimpleSave() method saves a book.

If you set the `interactive` parameter to `False` and you specify the book's current name for `saveAsName`, FrameMaker saves the book under its current name.

If you specify another filename for `saveAsName`, FrameMaker saves the book to that filename.

If you specify an empty string (""), FrameMaker does not save the file. Instead it sets `FA_errno` to `Constants.FE_BadParameter (-43)`.

If you set the `interactive` parameter to `True`, FrameMaker displays the Save dialog box and allows the user to choose a filename. The book's current name appears as the default file name.

The SimpleSave() method returns the book it saved, on success.

On failure, the method sets `FA_errno`, to one of the following values:

Error	Reason
<ul style="list-style-type: none"> <li>Constants.FE_Canceled (-44) Constants.FE_Canceled-ByClient (-86)</li> </ul>	<ul style="list-style-type: none"> <li>Constants.FV_FileNotWritable (33) - File was not writable.</li> <li>Constants.FV_BadSaveFileName (34) - The specified file name is not allowed by the operating system.</li> <li>Constants.FV_BadFileId (35) - The file's operating system ID was bad.</li> <li>Constants.FV_CancelSaveFileIsInUse (50) - The file is in use and the user did not or could not reset the lock. Or the file is in use, and the Constants.FS_FileIsInUse (17) parameter is set to Constants.FV_DoCancel (0), or it is set to Constants.FV_ResetLockAndContinue (7) but FrameMaker could not reset the lock.</li> <li>Constants.FV_CancelSaveModDateChanged (52) - The file has changed since the last time it was opened or saved in the current session.</li> <li>Constants.FV_LockCouldntBeReset (2) - File lock could not be reset.</li> <li>Constants.FV_LockWasReset (0) - File lock was reset.</li> <li>Constants.FV_LockNotReset (1) - File lock was not reset.</li> <li>Constants.FV_FileIsViewOnly (4) - File was View Only.</li> <li>Constants.FV_NonPortableSaveName (37) - File had a name that is not portable (Macintosh only).</li> <li>Constants.FV_NonPortableFileRefs (38) - Referenced graphics files had filenames that are not portable (Macintosh only).</li> </ul>
Constants.FE_WrongProduct (-60)	The specified file contains structure and the current product interface is not Structured FrameMaker.
Constants.FE_FailedState (-45) Constants.FE_BadParameter (-43)	The filename was invalid.
Constants.FE_FilterFailed (-89)	Constants.FV_InvalidSaveFilter (42) - The filter specified by Constants.FS_SaveFileTypeHint (16) is not installed, or the syntax for Constants.FS_SaveFileTypeHint (16) is invalid.

**Returns**[Book](#)**Syntax**

```
SimpleSave (saveAsName, interactive)
```

## Parameters

Parameter name	Data Type	Optional	Description
saveAsName	string	No	The absolute path in which to save the document or book.
interactive	int	No	Specifies whether FrameMaker displays messages and warnings to the user. Set this parameter to 1 to display messages and warnings.

## NewSeriesObject

### Description

The `NewSeriesObject()` method creates a series object. Series objects include the following object types:

- BodyPage
- BookComponent
- Pgf

The method allows you to specify the position in the series at which to add the new object.

The method returns the location of the new object, on success. If the method fails, it sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid book.
Constants.FE_BadNew (-23)	Object cannot be created.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_NotBodyPage (-24)	The <code>previous</code> parameter must specify the location of a body page.
Constants.FE_NotPgf (-25)	The <code>previous</code> parameter must specify the location of a paragraph.
Constants.FE_NotBookComponent (-26)	The <code>previous</code> parameter must specify the location of a book component.

### Returns

Object

**Syntax**

```
NewSeriesObject(objType, previous)
```

**Parameters**

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object to create (for example, FO_BodyPage or FO_Pgf).
previous	Object	No	The location of the object in the series after which to add the new object. To add a paragraph at the start of a flow, specify the location of the flow. To add an object at the beginning of any other series, specify 0.

**NewSeriesBookComponent****Description**

The `NewSeriesBookComponent()` method creates a Book Component.

The method allows you to specify the position in the series at which to add the new object.

See the [NewSeriesObject](#) method for more information.

**Returns**

[BookComponent](#)

**Syntax**

```
NewSeriesBookComponent(previous)
```

**Parameters**

Parameter name	Data Type	Optional	Description
previous	Object	No	The location of the object in the series after which to add the new Book Component. To add a paragraph at the start of a flow, specify the location of the flow. To add a Book Component at the beginning of any other series, specify 0.

## NewBookComponentOfTypeInHierarchy

### Description

Inserts a book component of a specified type at a specified position in a structured FrameMaker book.

### Returns

*BookComponent*

### Syntax

```
NewBookComponentOfTypeInHierarchy(compName, compType, elemLoc)
```

### Parameters

Parameter name	Data Type	Optional	Description
bookId	int	No	The ID of the book to add the component to
compName	string	No	The name of the component
elemLoc	<i>Element Loc</i>	No	The position at which to add the new element

**NOTE:** The book you specify for `bookId` must already be structured.

## GetConditionalSettings

### Description

Get a `PropVals` object containing the conditional settings of the current book.

### Returns

*PropVals*

### Syntax

```
GetConditionalSettings()
```

## ApplyConditionalSettings

### Description

Apply conditional settings in the selected book based on the specified settings.

**Returns**

Void

**Syntax**

ApplyConditionalSettings(settings)

**Parameters**

Parameter name	Data Type	Optional	Description
settings	<i>PropVals</i>	No	The value of the property to set.

The following are the list of settings can be applied:

Property	Meaning
Constants.FS_ShowState (1)	Specify the show state of the conditional text applied to the book. <ul style="list-style-type: none"> <li>Constants.FV_ShowAll (1)</li> <li><i>Constants.FV_ShowAsPerConditions (2)</i></li> <li><i>Constants.FV_ShowAsPerExpression (3)</i></li> </ul>
Constants.FS_ShowConditions (2)	A string array of the names of the condition tags to be added to the Show list in the Show/Hide Conditional Text dialog
Constants.FS_HideConditions (3)	A string array of the names of the condition tags to be added to the Hide list in the Show/Hide Conditional Text dialog
Constants.FS_ActiveConditionalExpression (6)	The conditional build expression tag to be applied. <ul style="list-style-type: none"> <li>When reading this property: Returns the currently applied Conditional Build Expression tag, if (FS_ShowState = FV_ShowAsPerExpression). Else NULL is returned.</li> </ul> <p><i>When applying this property: Applies the current active Conditional Build expression if (FS_ShowState = FV_ShowAsPerExpression). Otherwise ignored.</i></p>
Constants.FS_ShowConditionIndicators (7)	Show / hide the conditional indicators.
Constants.FS_ApplyConditionalSettingsToViewOnlyDoc (8)	Apply the conditions to view-only documents in the book.

Property	Meaning
Constants.FS_ApplyConditionalSettingsToNestedBooks (9)	Apply the conditions to nested books within the current book.
Constants.FS_ApplyConditionalSettingsShowBookErrorLog (10)	Show errors in the book error log.

The method returns FE\_Success if it succeeds. On failure, the method returns one of the following values to FA\_errno:

Error	Reason
Constants.FE_ExpressionNotFound	Expression Tag to be applied does not exist in the default document of the book.
Constants.FE_FailedToApplyOnOneOrMoreComponents	Failed to apply conditional settings on one or more book components.

## ManageConditionalExpressions

### Description

Add, edit, or delete conditional expression tags to the current book. Applies to the options available in the Add/Edit Conditional Tag dialog.

### Returns

Void

### Syntax

```
ManageConditionalExpressions (settings)
```

### Parameters

Parameter name	Data Type	Optional	Description
settings	<i>PropVals</i>	No	The value of the property to set.

Use the following are add, edit, and delete settings:

Property	Meaning
Constants.FS_AddEditExpressions (1)	String array of pair of strings (expression tag, expression definition) to be added/edited in conditional expression catalog.
Constants.FS_DeleteExpressions (2)	String array of expression tags to be deleted from conditional expression catalog.

The method returns FE\_Success if it succeeds. On failure, the method returns one of the following values to FA\_erno:

Error	Reason
Constants.FE_CannotAddEditExpressionsInOneOrMoreComponents	Failed to Add/Edit the Expression on one or more book components while using F_ApiManageConditionalExpressions(). See FrameMaker Error pod for more details.

## GetConditionalExpression

### Description

Returns the conditional expression definition for the given named expression.

### Returns

exprName

### Syntax

GetConditionalExpression(exprName)

### Parameters

Parameter name	Data Type	Optional	Description
exprName	String	No	Name of the required conditional build expression.



## Save

### Description

The `Save()` method saves a book. The method allows you to script the way FrameMaker saves the file and to specify responses to warnings and messages that arise while the file is being saved.

You can save a file under its current name or save it as a new file.

The property list for the `saveParams` parameter is as follows:

Property	Meaning
<code>Constants.FS_SavedFileName</code> (1)	A string that specifies the full path of the saved file.
<code>Constants.FS_SaveNativeError</code> (2)	The error condition. If the file is saved successfully, this property is set to <code>FE_Success</code> .
<code>Constants.FS_SaveStatus</code> (3)	A bit field indicating what happened when the file was saved.

The method returns the document it saved, on success.

On failure the method sets `FA_errno` to one of the following values:

Error	Reason
<ul style="list-style-type: none"> <li>Constants.FE_Canceled (-44) Constants.FE_Canceled-ByClient (-86)</li> </ul>	<ul style="list-style-type: none"> <li>Constants.FV_FileNotWritable (33) - File was not writable.</li> <li>Constants.FV_BadSaveFileName (34) - The specified filename is not allowed by the operating system.</li> <li>Constants.FV_BadFileId (35) - The file's operating system ID was bad.</li> <li>Constants.FV_CancelSaveFileIsInUse (50) - The file is in use and the user did not or could not reset the lock. Or the file is in use, and the Constants.FS_FileIsInUse (17) parameter is set to Constants.FV_DoCancel (0), or it is set to Constants.FV_ResetLockAndContinue (7) but FrameMaker could not reset the lock.</li> <li>Constants.FV_CancelSaveModDateChanged (52) - The file has changed since the last time it was opened or saved in the current session.</li> <li>Constants.FV_LockCouldntBeReset (2) - File lock could not be reset.</li> <li>Constants.FV_LockWasReset (0) - File lock was reset.</li> <li>Constants.FV_LockNotReset (1) - File lock was not reset.</li> <li>Constants.FV_FileIsViewOnly (4) - File was View Only.</li> <li>Constants.FV_NonPortableSaveName (37) - File had a name that is not portable (Macintosh only).</li> <li>Constants.FV_NonPortableFileRefs (38) - Referenced graphics files had filenames that are not portable (Macintosh only).</li> </ul>
Constants.FE_WrongProduct (-60)	The specified file contains structure and the current product interface is not Structured FrameMaker.
Constants.FE_FailedState (-45) Constants.FE_BadParameter (-43)	The filename was invalid.
Constants.FE_FilterFailed (-89)	Constants.FV_InvalidSaveFilter (42) - The filter specified by Constants.FS_SaveFileTypeHint (16) is not installed, or the syntax for Constants.FS_SaveFileTypeHint (16) is invalid.

**Returns**

Object

**Syntax**

```
Save (saveAsName, saveParams, saveReturnParams)
```

## Parameters

Parameter name	Data Type	Optional	Description
saveAsName	string	No	The path for saving the document or book.
saveParams	<i>PropVals</i>	No	A property list that tells FrameMaker how to save the file and how to respond to errors and other conditions. Use <code>GetSaveDefaultParams()</code> or <code>AllocatePropVals()</code> to create and allocate memory for this property list. To use the default list, specify <code>null</code> .
saveReturnParams	<i>PropVals</i>	No	A property list that returns information about how the FrameMaker product saved the file.

## UpdateBook

### Description

The `UpdateBook()` method updates a book.

The method allows you to specify a script (property list) specifying how to update the book and how to deal with error and warning conditions.

For example, you can specify whether to abort or to continue updating a book if it contains view-only documents.

**NOTE:** Always initialize the pointer to the property list that you specify for `updateReturnParams` to null before you call `UpdateBook()`.

To get a property list to specify for the `updateParams` parameter, use `GetUpdateBookDefaultParams()` or create the list from scratch.

The following are the properties that you can use in the property list:

Property	Meaning
<code>Constants.FS_AlertUserAboutFailure</code> (2)	Alert user with warnings and messages if necessary. Set this parameter to True or False as desired. <ul style="list-style-type: none"> <li>• True - Notify user when unexpected conditions occur.</li> <li>• False - Do not notify user when unexpected conditions occur.</li> </ul>

Property	Meaning
Constants.FS_AllowInconsistentNum Props (6)	<p>Allow the FrameMaker product to update numbering, text insets and other properties of all the FrameMaker documents in the book, even if there are documents in the book with numbering properties that do not match the properties specified in the book. The permissible values are as follows:</p> <ul style="list-style-type: none"> <li>Constants.FV_DoOK (1) - Update numbering even if there are inconsistent properties in the book.</li> </ul> <p><i>Constants.FV_DoCancel (0) - Cancel the update operation when FrameMaker encounters a document with inconsistent numbering properties.</i></p> <p><i>Constants.FV_DoShowDialog (4) - Show a dialog box and let the user decide.</i></p>
Constants.FS_AllowNonFMFiles (1)	<p>Allow the FrameMaker product to update numbering, text insets, and other properties of all the FrameMaker documents in the book, even if there are documents in the book that were not created by FrameMaker. The permissible values are as follows:</p> <ul style="list-style-type: none"> <li>Constants.FV_DoOK (1) - Update the book even if the book contains files not created by FrameMaker.</li> </ul> <p><i>Constants.FV_DoCancel (0) - Cancel the update operation when it encounters a document not created by FrameMaker.</i></p> <p><i>Constants.FV_DoShowDialog (4) - Show a dialog box and let the user decide.</i></p>
Constants.FS_AllowViewOnlyFiles (3)	<p>Allow the FrameMaker product to update view-only documents in the book. The permissible values are as follows:</p> <ul style="list-style-type: none"> <li>Constants.FV_DoOK (1) - Update the view-only documents.</li> </ul> <p><i>Constants.FV_DoCancel (0) - Cancel the entire update operation when it encounters a view-only document.</i></p> <p><i>Constants.FV_DoShowDialog (4) - Show a dialog box and let the user decide.</i></p>
Constants.FS_MakeVisible (5)	<p>Make newly generated files (lists and indexes) visible. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>True - Make visible.</li> </ul> <p><i>False - Do not make visible.</i></p>

Property	Meaning
Constants.FS_ShowBookErrorLog (4)	<p>Display the book error log for this update operation. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>False - Do not display the error log; all warnings and errors are redirected to the console.</li> </ul> <p><i>True - Display the error log.</i></p>
Constants.FS_UpdateBookGenerated Files (7)	<p>Update generated files such as TOC, lists, and indexes. Only update those generated files that have GenerateInclude set to True in their associated FO_BookComponent objects. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>True - Update generated files.</li> </ul> <p><i>False - Do not update generated files.</i></p>
Constants.FS_UpdateBookMasterPages (12)	<p>Update automatic assignment of master pages in all the documents of the book. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>True - Update master pages.</li> </ul> <p><i>False - Do not update master pages.</i></p>
Constants.FS_UpdateBookNumbering (8)	<p>Update numbering in all the documents of the book. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>True - Update numbering.</li> </ul> <p><i>False - Do not update numbering.</i></p>
Constants.FS_UpdateBookOleLinks (9)	<p>Update OLE links in all the documents of the book. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>True - Update OLE links.</li> </ul> <p><i>False - Do not update OLE links.</i></p>
Constants.FS_UpdateBookTextReferences (10)	<p>Update text insets in all the documents of the book. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>True - Update text insets.</li> </ul> <p><i>False - Do not update text insets.</i></p>
Constants.FS_UpdateBookXRefs (11)	<p>Update cross-references in all the documents of the book. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>True - Update cross-references.</li> </ul> <p><i>False - Do not update cross-references.</i></p>

The property list to which `updateReturnParams` is set, has the following property:

Property	Meaning
Constants.FS_UpdateBookStatus (1)	A bit field to indicate what happened during the update.

To determine if a particular `Constants.FS_UpdateBookStatus (1)` bit is set, use `CheckStatus()`.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid book
Constants.FE_BadOperation ()	<ul style="list-style-type: none"> <li>Constants.FV_BookNotSelfConsistent (0) - The book is not self-consistent (book generates data in one file that is source data for another generated file, or page count continually changes for this operation). <i>Constants.FV_DuplicateFileInBook (1) - One or more files in the book is a duplicate of another file.</i> <i>Constants.FV_NoNonGeneratedFilesInBook (2) - The only files in the book are generated files.</i></li> </ul>
Constants.FE_BadParameter (-43)	<ul style="list-style-type: none"> <li>Constants.FV_BadUpdateBookFileId (64) - Specified book is invalid. <i>Constants.FV_BadUpdateBookScriptValue (65) - The update book script contains an invalid property value.</i></li> </ul>
Constants.FE_Canceled (-44) Constants.FE_CanceledByClient (-86)	<ul style="list-style-type: none"> <li>Constants.FV_CancelInconsistentNumPropsInFileInBook (34) - One or more of the book's document files has numbering properties that are inconsistent with the properties stored in the book. <i>Constants.FV_CancelNonFMFileInBook (32) - One or more of the book's document files is not a FrameMaker file.</i> <i>Constants.FV_CancelViewOnlyFileInBook (33) - One or more of the book's document files is view-only.</i> <i>Constants.FV_UserCanceledUpdateBook (35) - The user canceled the update operation.</i></li> </ul>

Error	Reason
Constants.FE_SystemError (-42)	<ul style="list-style-type: none"> <li>Constants.FV_FileInBookNotOpened (96) - One or more files in the book could not be opened.</li> <li>Constants.FV_FileInBookNotSaved (97) - One or more files in the book could not be saved.</li> <li>Constants.FV_TooManyWindowsUpdateBook (98) - Not enough memory. Too many windows were open.</li> </ul>

**Returns**

int

**Syntax**

UpdateBook(updateParams, updateReturnParams)

**Parameters**

Parameter name	Data Type	Optional	Description
updateParams	<i>PropVals</i>	No	A property list specifying how to update the book and how to respond to errors and other conditions. To use the default list, specify <code>null</code> .
updateReturnParams	<i>PropVals</i>	No	A property list that provides information about how the FrameMaker product updated the book. This parameter must be initialized before you call <code>UpdateBook()</code> .

**Import****Description**

The `Import()` method imports text or graphics into a document.

`Import()` allows you to specify a script (property list) specifying how to import text or graphics and how to deal with error and warning conditions.

For example, you can specify whether to import a file by reference or by copy.

If you import a file by reference, `Import()` creates an inset. The following list summarizes the types of files you can import with `Import()` and the types of inset objects it creates when you import them by reference:

- Graphics - `FO_Inset`

- Text - FO\_TiText, FO\_TiTextTable
- Frame binary document - FO\_TiFlow
- MIF - FO\_TiFlow

**NOTE:** When importing a graphic, you can specify that it be imported at its default resolution by setting the `Constants.FS_GraphicDpi` (15) property to 0 and setting the `Constants.FS_FitGraphicInSelectedRect` (14) property to `False`. If the graphic has no default resolution, it is imported at 72 dpi

To get a property list to specify for the `importParams` parameter, use `GetImportDefaultParams()` or create the list from scratch.

**NOTE:** Always initialize the pointer to the property list that you specify for `importReturnParams` to null before you call `Import()`.

The following are the list of properties that can be imported:

Property	Meaning
<code>Constants.FS_AlertUserAboutFailure</code> (2)	Alert user if an unexpected condition, such as an unrecognized file type, occurs. Set to either True or False <ul style="list-style-type: none"> <li>• True - Notify user when unexpected conditions occur.</li> <li><i>False - Do not notify user when unexpected conditions occur.</i></li> </ul>
<code>Constants.FS_DisallowDoc</code> (8)	Disallow importing FrameMaker binary documents. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Do not allow importing FrameMaker binary documents.</li> <li><i>False - Allow importing FrameMaker binary documents.</i></li> </ul>
<code>Constants.FS_DisallowFilterTypes</code> (11)	Disallow importing filterable files. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Do not allow importing filterable files.</li> <li><i>False - Allow importing filterable files.</i></li> </ul>
<code>Constants.FS_DisallowMIF</code> (7)	Disallow importing MIF files. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Do not allow importing MIF files.</li> <li><i>False - Allow importing MIF files.</i></li> </ul>
<code>Constants.FS_DisallowGraphicTypes</code> (9)	Disallow importing graphic files. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Do not allow importing graphic files.</li> <li><i>False - Allow importing graphic files.</i></li> </ul>
<code>Constants.FS_DisallowMacEditions</code> (10)	Disallow importing Macintosh editions of files. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Do not allow importing Macintosh editions of files.</li> <li><i>False - Allow importing Macintosh editions of files.</i></li> </ul>



Property	Meaning
Constants.FS_DisallowPlainText (12)	Disallow importing Text Only files. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Do not allow importing Text Only files.</li> <li>• False - Allow importing Text Only files.</li> </ul>
Constants.FS_DisallowSgml (33)	Disallow importing SGML documents. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Do not allow importing SGML documents.</li> <li>• False - Allow importing SGML documents.</li> </ul>
Constants.FS_DisallowXml (43)	Disallow importing XML documents. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Do not allow importing XML documents.</li> <li>• False - Allow importing XML documents.</li> </ul>
Constants.FS_DontNotifyAPIClients (3)	Notify other clients of the import operation. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Do not notify other clients of the import operation.</li> <li>• False - Notify other clients of the import operation.</li> </ul>
Constants.FS_FileTypeHint (34)	If the file is filterable, this property is a string that enables FrameMaker to automatically call the correct filter to filter the file. If the file is not filterable, this property contains the value <code>null</code> .
Constants.FS_FileIsSgmlDoc (32)	File is an SGML document. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_DoOK (1) - Import it anyway.</li> <li>• Constants.FV_DoCancel (0) - Cancel the import operation.</li> <li>• Constants.FV_DoShowDialog (4) - Show a dialog box and let the user decide.</li> </ul>
Constants.FS_FileIsXmlDoc (37)	File is an XML document. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_DoOK (1) - Import it anyway.</li> <li>• Constants.FV_DoCancel (0) - Cancel the import operation.</li> <li>• Constants.FV_DoShowDialog (4) - Show a dialog box and let the user decide.</li> </ul>
Constants.FS_ForceImportAsText (17)	Import the file as a Text Only document, even if it is a MIF file or a filterable file. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Import it as a Text Only file.</li> <li>• False - Import it in a format based on its type.</li> </ul>

Property	Meaning
Constants.FS_HowToImport (4)	Import file by reference or copy. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_DoByRef (5) - Import file by reference.</li> <li><i>Constants.FV_DoByCopy (6) - Import file by copy.</i></li> <li><i>Constants.FV_DoUserChoice (7) - Allow user to choose how to import the file.</i></li> </ul>
Constants.FS_ImportAsType (36)	Specify the format of the file to import. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_AUTORECOGNIZE (0x00) - Default value; recognize the file type automatically.</li> <li><i>Constants.FV_TYPE_BINARY (0x01) - A FrameMaker binary file.</i></li> <li><i>Constants.FV_TYPE_MIF (0x02) - A MIF file.</i></li> <li><i>Constants.FV_TYPE_TEXT (0x03) - A file containing plain text.</i></li> <li><i>Constants.FV_TYPE_SGML (0x04) - An SGML file.</i></li> <li><i>Constants.FV_TYPE_XML (0x05) - An XML file.</i></li> <li><i>Constants.FV_TYPE_FILTER (0xFF) - Use a filter to import this file. You must specify a valid file type hint for the Constants.FS_FileTypeHint (34) property.</i></li> </ul>
Constants.FS_ManualUpdate (5)	Update inset manually. Set to either True or False. <ul style="list-style-type: none"> <li>True - Update inset automatically.</li> <li><i>False - Do not update inset manually.</i></li> </ul>
Constants.FS_SgmlImportApplication (35)	Retained for compatibility with older versions of FrameMaker. Use the Constants.FS_StructuredImportApplication (35) property instead.
Constants.FS_StructuredImportApplication (35)	String specifying the name of the structure application to use when importing an XML or SGML file. This property takes precedence over any other structure application specification. If the specified application does not exist, the calling function will fail. If this property is not set, then the value of this property is <code>null</code> .
Constants.FS_ShowBrowser (1)	Display the Import dialog box. Set to either True or False. <ul style="list-style-type: none"> <li>True - Display the Import dialog box.</li> <li><i>False - Do not display the Import dialog box.</i></li> </ul>
Constants.FS_TextInsetName (6)	Denotes the Inset name. If not set, the value of this property is <code>null</code> .

The syntax of the string passed to the `Constants.FS_FileTypeHint` (34) is as follows:

```
record_vers vendor format_id platform filter_vers filter_name
```

**NOTE:** The fields in the record are not separated by spaces. Spaces are significant data except those that appear in the `filter_name` field.

For example:

```
0001PGRFPICMAC61.0 Built-in PICT reader0001ORBLSVGIA11 Platform SVG
reader for 7.0
```

Each field of the record (except `filter_name`) specifies a four-byte code. If a code contains fewer than four alphanumeric characters, the remaining bytes must be filled out with spaces.

`record_vers` specifies the version of the record, currently 0001.

`vendor` is a code specifying the filter's vendor. The code is a string of four characters. The following table lists the possible codes

Code	Meaning
PGRF	Built-in Frame filters.
FAPI	External Frame FDK client filter.
FFLT	External Frame filters.
IMAG	External ImageMark filters.
XTND	External XTND filters.

`format_id` is a code specifying the format that the filter translates. The code is a string of four characters. The following are the possible codes:

Code	Meaning
CDR	CorelDRAW
CGM	Computer Graphics Metafile
DIB	Device-independent bitmap (Windows)
DRW	Micrografx CAD
DXF	Autodesk Drawing eXchange file (CAD files)
EMF	Enhanced Metafile (Windows)
EPSB	Encapsulated PostScript Binary (Windows)
EPSD	Encapsulated PostScript with Desktop Control Separations (DCS)
EPSF	Encapsulated PostScript (Macintosh)
EPSI	Encapsulated PostScript Interchange

Code	Meaning
FRMI	FramedImage
FRMV	FrameVector
G4IM	CCITT Group 4 to Image
GEM	GEM file (Windows)
GIF	Graphics Interchange Format (CompuServe)
HPGL	Hewlett-Packard Graphics Language
IGES	Initial Graphics Exchange Specification (CAD files)
IMG4	Image to CCITT Group 4 (UNIX)
MooV	QuickTime Movie
OLE	Object Linking and Embedding Client (Microsoft)
PCX	PC Paintbrush
PICT	QuickDraw PICT
PNTG	MacPaint
SNRF	Sun Raster File
SRGB	SGI RGB
SVGI	Scalable Vector Graphics
TIFF	Tag Image File Format
WMF	Windows Metafile
WPG	WordPerfect Graphics
XWD	X Windows System Window Dump file

`platform` is a code specifying the platform on which the filter was run. The code is a string of four characters. The following are the possible codes:

Code	Meaning
MAC6	Macintosh 68000 series
MACP	Power Macintosh
WINT	Windows NT
WIN3	Windows 3.1

Code	Meaning
WIN4	Windows 95
OS/2	IBM OS/2
UNIX	Generic X/11 (Sun, HP)

`filter_vers` is a string of four characters identifying the version of the filter on that platform. For example, version 1.0 of a filter is represented by the string 1.0.

`filter_name` is a text string (up to 31 characters long) that describes the filter.

`Import()` uses the following properties only for importing `FrameMaker` documents and MIF files:

Property	Meaning
<code>Constants.FS_FileIsMakerDoc (18)</code>	File is a FrameMaker binary document or a MIF file. The possible values are: <ul style="list-style-type: none"> <li><code>Constants.FV_DoOK (1)</code> - Import it anyway.</li> <li><code>Constants.FV_DoCancel (0)</code> - Cancel the import operation.</li> <li><code>Constants.FV_DoShowDialog (4)</code> - Show a dialog box and let the user decide.</li> </ul>
<code>Constants.FS_FormatImportedText (22)</code>	Format the imported text. The possible values are: <ul style="list-style-type: none"> <li><code>Constants.FV_EnclosingDoc (1)</code> - Use formatting in the enclosing document.</li> <li><code>Constants.FV_PlainText (2)</code> - Format the imported text as plain text.</li> <li><code>Constants.FV_SourceDoc (0)</code> - Use formatting from the source documents.</li> </ul>
<code>Constants.FS_ImportFlowPageSpace (21)</code>	If <code>Constants.FS_UseMainFlow (19)</code> is <code>False</code> , then <code>Constants.FS_ImportFlowPageSpace (21)</code> indicates the type of pages to search for the flow specified by the <code>Constants.FS_ImportFlowTag (20)</code> property. The possible values are: <ul style="list-style-type: none"> <li><code>Constants.FV_BodyPage (10)</code> - Search body pages.</li> <li><code>Constants.FV_ReferencePage (9)</code> - Search reference pages.</li> </ul>
<code>Constants.FS_ImportFlowTag (20)</code>	If <code>Constants.FS_UseMainFlow (19)</code> is <code>False</code> , then <code>Constants.FS_ImportFlowTag (20)</code> indicates the name of the flow to import. If this value is not set, the default value is <code>null</code> .

Property	Meaning
Constants.FS_RemoveManualPageBreaks (23)	Remove manual page breaks if FS_FormatImportedTest is set to Constants.FV_EnclosingDoc (1). Set to either True or False. <ul style="list-style-type: none"> <li>True - Remove manual page breaks.</li> </ul> <i>False - Do not remove manual page breaks.</i>
Constants.FS_RemoveOverrides (24)	Remove format overrides if Constants.FS_FormatImportedTest is set to Constants.FV_EnclosingDoc (1). Set to either True or False. <ul style="list-style-type: none"> <li>True - Remove format overrides.</li> </ul> <i>False - Do not remove format overrides.</i>
Constants.FS_UseMainFlow (19)	Import text from specified document's main flow. Set to either True or False. <ul style="list-style-type: none"> <li>True - Import the text from the main flow.</li> </ul> <i>False - Do not import the text from the main flow.</i>

Import() uses the following properties only for importing graphics files:

Property	Meaning
Constants.FS_FileIsGraphic (13)	File is a graphic file. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_DoOK (1) - Import it.</li> </ul> <i>Constants.FV_DoCancel (0) - Cancel import operation.</i> <i>Constants.FV_DoShowDialog (4) - Display a dialog box and let the user decide.</i>
Constants.FS_FitGraphicInSelectedRect (14)	Fit the graphic in the selected graphic frame. Set to either True or False. <ul style="list-style-type: none"> <li>True - Fit the graphic in the frame.</li> </ul> <i>False - Do not fit the graphic in the frame.</i>
Constants.FS_GraphicDpi (15)	Integer specifying dots per inch (DPI) at which to import the graphic. The default value is 72 (to specify 72 DPI).
Constants.FS_InsetRasterDpi	Denotes the read-only value to indicate the DPI of the bitmap facet when importing SVG data.
Constants.FS_RasterDpi (38)	Specifies the DPI for the bitmap facet when importing SVG data. This corresponds to any of the DPI settings in the Imported Graphic Scaling dialog box.

Property	Meaning
Constants.FS_RasterImageHeight (40)	Specifies the height of the rectangle in which to import the graphic, when importing SVG data. <i>NOTE: You should always specify this value in points. This property corresponds to the height for the Fit in Selected Rectangle setting in the Imported Graphic Scaling dialog box.</i>
Constants.FS_RasterImageWidth (39)	Specifies the width of the rectangle in which to import the graphic, when importing SVG data. <i>NOTE: You should always specify this value in points. This property corresponds to the width for the Fit in Selected Rectangle setting in the Imported Graphic Scaling dialog box.</i>
Constants.FS_ShowRasterDpiDialog (41)	For importing SVG data, show the Imported Graphic Scaling dialog box. Default is 1 (one). The values are: <ul style="list-style-type: none"> <li>1 (one) or any non-zero value: Display the Imported Graphic Scaling dialog box.</li> </ul> <i>0 (zero): Do not display the Imported Graphic Scaling dialog box.</i>

`Import()` uses the following properties only for importing ASCII text files:

Property	Meaning
Constants.FS_CellSeparator (29)	If Constants.FS_FileIsText (16) is set to Constants.FV_DoImportAsTable (8), then this property denotes the delimiter or separator used to parse the text into cells. The default value is <code>null</code> .
Constants.FS_FileIsText (16)	File is a Text Only file. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_TextFile_EOLisEOP (12) - Import the file and convert each end-of-line into a paragraph break.</li> </ul> <i>Constants.FV_TextFile_EOLisNotEOP (13) - Import the file but do not convert each end-of-line into a paragraph break.</i> <i>Constants.FV_DoImportAsTable (8) - Import the file into a table.</i> <i>Constants.FV_DoCancel (0) - Cancel the import operation.</i>
Constants.FS_ImportTblTag (25)	Specifies the table format to use, if Constants.FS_FileIsText (16) is set to Constants.FV_DoImportAsTable (8). The default value is <code>null</code> .

Property	Meaning
Constants.FS_LeaveHeadingRows Empty (27)	<p>Leave the heading rows empty if Constants.FS_FileIsText (16) is set to Constants.FV_DoImportAsTable (8). Constants.FS_LeaveHeadingRowsEmpty (27) is set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Leave heading rows empty.</li> </ul> <p><i>False - Do not leave heading rows empty.</i></p>
Constants.FS_NumCellSeparators (30)	<p>Denotes the number of spaces to use as a separator, if Constants.FS_FileIsText (16) is set to Constants.FV_DoImportAsTable (8), and Constants.FS_CellSeparator (29) is set to a space (' '). The default value is 1.</p>
Constants.FS_NumColumns (31)	<p>Specifies the number of columns in the table, if Constants.FS_FileIsText (16) is set to Constants.FV_DoImportAsTable (8), and Constants.FS_TreatParaAsRow (28) is False. The default value is 1.</p>
Constants.FS_TblNumHeadingRows (26)	<p>Specifies the number of heading rows in the table, if Constants.FS_FileIsText (16) is set to Constants.FV_DoImportAsTable (8). The default value is 1.</p>
Constants.FS_TreatParaAsRow (28)	<p>Convert each line in the text file into a row of table cells and use Constants.FS_CellSeparator (29) and Constants.FS_NumCellSeparators (30) to determine how to divide the line into separate cells, if Constants.FS_FileIsText (16) is set to Constants.FV_DoImportAsTable (8). Constants.FS_TreatParaAsRow (28) is set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Convert each line into a row of table cells.</li> </ul> <p><i>False - Convert each line into a table cell instead.</i></p>

When you import text into a table, in addition to setting `Constants.FS_FileIsText (16)` to `Constants.FV_DoImportAsTable (8)`, you must specify a value for the `Constants.FS_ImportTblTag (25)` property. If you set the `Constants.FS_TreatParaAsRow (28)` property to `True`, you must also specify a value for the `Constants.FS_CellSeparator (29)` property.

The property list returned by `GetImportDefaultParams()` does not specify values for the `Constants.FS_ImportTblTag (25)` and `Constants.FS_CellSeparator (29)` properties.

If you use the property list to import a table and do not specify a value for `Constants.FS_ImportTblTag (25)`, `Import()` fails and sets `FA_errno` to `Constants.FE_BadParameter (-43)`.



If you set `Constants.FS_TreatParaAsRow (28)` to `True` and do not specify a cell separator by setting `Constants.FS_CellSeparator (29)`, `Import ()` fails and sets `FA_errno` to `Constants.FE_BadParameter (-43)`.

If `GetImportDefaultParams ()` fails, the `Import ()` method sets the `len` field of the returned `PropVals` structure to 0.

The `Import ()` method returns the object that was imported, on success.

**NOTE:** `Import()` returns 0 on success, if it imports a copy or a graphic inset.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_SystemError (-42)</code>	System error, such as an unreadable file or insufficient memory.
<code>Constants.FE_BadParameter (-43)</code>	The property list contained an invalid parameter.
<code>Constants.FE_BadFileType (-83)</code>	The specified file exists, but it does not have the correct file type.
<code>Constants.FE_MissingFile (-84)</code>	The specified file does not exist.
<code>Constants.FE_NoSuchFlow (-82)</code>	The script specifies an import flow that does not exist.
<code>Constants.FE_FailedState (-45)</code>	Internal error.
<code>Constants.FE_CircularReference (-81)</code>	Importing the specified file causes a circular reference.
<code>Constants.FE_FileClosedByClients</code>	The file was closed by a client before it could be imported.

The property list returned to `importReturnParamspp` has the following properties:

Property	Meaning
<code>Constants.FS_ImportedFileName (1)</code>	A string specifying the source file's path. If you scripted <code>Constants.FS_ShowBrowser (1)</code> , this path can be different from the one you specified in the <code>Import</code> script.
<code>Constants.FS_ImportNativeError (2)</code>	The error condition; normally the same value as <code>FA_errno</code> . If the file is imported successfully, <code>Constants.FS_ImportNativeError (2)</code> is set to <code>Constants.FE_Success (0)</code> .

Property	Meaning
Constants.FS_ImportStatus (3)	A bit field indicating what happened when the file was imported. See the following table for more information.

Both the `Constants.FS_ImportNativeError` (2) property and the `FA_errno` global variable indicate the result of a call to the `Import()` method. The following table lists the possible status flags and the reasons associated with them:

Status Flag	Reason
<ul style="list-style-type: none"> <li>Constants.FE_BadParameter (-43)</li> <li>Constants.FE_BadFileType (-83)</li> <li>Constants.FE_MissingFile (-84)</li> <li>Constants.FE_FailedState (-45)</li> <li>Constants.FE_CanceledByClient (-86) (file was not imported)</li> </ul>	<ul style="list-style-type: none"> <li>Constants.FV_BadImportFileName (16) - The specified source filename is invalid.</li> <li>Constants.FV_BadImportFileType (17) - The Import script specified a file type different from the source file's actual type.</li> <li>Constants.FV_BadImportScriptValue (18) - The Import script contained an invalid property value.</li> <li>Constants.FV_BadTextFileTypeHint (29) - The file was a text file, and the string in <code>Constants.FS_FileTypeHint</code> (34) was not a valid import hint string.</li> <li>Constants.FV_MissingScript: <code>Import()</code> was called without a script.</li> <li>Constants.FV_DisallowedImportType (21) - Source file's type is disallowed by script.</li> <li>Constants.FV_NoMainFlow (22) - A script was specified to import the main flow, but the source file does not have a main flow.</li> <li>Constants.FV_NoFlowWithSpecifiedName (23) - A script was specified with a flow name that does not exist.</li> <li>Constants.FV_InsertionPointNotInText (24) - The insertion point in the enclosing document is not in the text.</li> <li>Constants.FV_InsufficientMemory (27) - There is insufficient memory to import the source file.</li> <li>Constants.FV_BadEnclosingDocId (28) - There is no open document with the specified ID.</li> <li>Constants.FV_ImportFileNotReadable (48) - The specified source file is unreadable.</li> </ul>

Status Flag	Reason
Constants.FE_Success (0)	<ul style="list-style-type: none"> <li>• Constants.FV_ImportedByCopy (0) - The source file was imported by copy.  <i>Constants.FV_ImportTextTable</i> - The source file is a text file, which was imported into a table.  <i>Constants.FV_ImportedMIF (3)</i> - The source file is a MIF file.  <i>Constants.FV_ImportedMakerDoc (4)</i> - The source file is a FASL file.  <i>Constants.FV_ImportedFilteredFile (5)</i> - The source file was filtered.  <i>Constants.FV_ImportedGraphicFile (6)</i> - The source file is a graphics file.  <i>Constants.FV_ImportedSgmlDoc (8)</i> - The source file is an SGML document.  <i>Constants.FV_ImportedXmlDoc (9)</i> - The source file is an XML document.</li> </ul>

Status Flag	Reason
Constants.FE_Canceled (-44)	<ul style="list-style-type: none"> <li>           • Constants.FV_CancelFileText (32) - The file is text, so the user or the Import script canceled the Import operation.  <i>Constants.FV_CancelFileGraphic (36) - The source file is a graphic, so the user or the Import script canceled the Import operation.</i>  <i>Constants.FV_CancelFileMacEdition - The source file is a Macintosh Edition, so the Import script canceled the Import operation</i>  <i>Constants.FV_CancelFileDoc (33) - The file is a FASL file, so the user or the script canceled the Import operation.</i>  <i>Constants.FV_CancelFileSgml (40) - The file is an SGML document, so the user or the script canceled the Import operation</i>  <i>Constants.FV_CancelFileXml (41) - The file is an XML document, so the user or the script canceled the Import operation.</i>  <i>Constants.FV_CancelFileMIF (34) - The source file is a MIF file, so the user or the script canceled the Import operation.</i>  <i>Constants.FV_CancelFileFilterable (35) - The source file is a filterable file, so the user or the script canceled the Import operation.</i>  <i>Constants.FV_InsertionPointInFootnote (26) - The insertion point was in a footnote and the import script specified to import the file as a table, so the file could not be imported.</i>  <i>Constants.FV_InsertionPointInTableCell (25) - The insertion point was in a table cell and the import script specified to import the file as a table, so the file could not be imported</i>  <i>Constants.FV_UserCanceledImport (37) - The user canceled the Import operation.</i>  <i>Constants.FV_UserCanceledImportBrowser : The user canceled the Import browser</i> </li> </ul>

To determine whether a particular `Constants.FS_ImportStatus` (3) bit is set, use `CheckStatus()`.

### Returns

Object

## Syntax

```
Import(textLocP, filename, importParams, importReturnParams)
```

## Parameters

Parameter name	Data Type	Optional	Description
textLocP	<i>TextLoc</i>	No	The text location at which to import the file.
filename	string	No	The full path of the file to import.
importParams	<i>PropVals</i>	No	A property list specifying how to import the file and how to respond to errors and other conditions. To use the default list, specify null.
importReturnParams	<i>PropVals</i>	No	A property list that provides information about how FrameMaker imported the file. This parameter must be initialized before you call Import().

## GetProps

### Description

Retrieves the properties of the specified Book.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the value of the specified property of the Book.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

## Syntax

```
SetProps (setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The value to set.

## ObjectValid

### Description

Indicates whether the specified book is valid. Returns 1 if the specified book is valid, else returns 0.  
See [ObjectValid](#) under the AFrame class for more information.

### Returns

```
int
```

### Syntax

```
ObjectValid()
```

## BookComponent

### Delete

#### Description

Deletes the specified BookComponent object.  
See [Delete](#) under the AFrame class for more information.

#### Returns

```
int
```

#### Syntax

```
Delete()
```

## MoveComponent

### Description

Moves a particular book component up and down in the book components sequence. In addition , you can promote and demote operations of a component in a hierarchy using this method.

### Returns

Void

### Syntax

```
MoveComponent (moveAction)
```

### Parameters

Parameter name	Data Type	Optional	Description
moveAction	int	No	Specifies the action to move the component

## GetProps

### Description

Retrieves the properties of the specified BookComponent object.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the property of the BookComponent object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

## Syntax

```
SetProps (setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The value of the property to be set.

## ObjectValid

### Description

Indicates whether a BookComponent object is valid. Returns 1 if the specified book component is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Cell

### StraddleCells

#### Description

The `StraddleCells()` method straddles the specified cells in a table.

The cells you straddle must all be from the same type of row. You cannot straddle a set of cells that are in both heading and body rows or footing and body rows. Also, the cells you straddle must be unstraddled. You cannot use this function to further straddle cells that are already straddled.

The `StraddleCells()` method returns `FE_Success` on success.

The method assigns one of the following values to `FA_errno`, on failure:



Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker product does not support tables.
Constants.FE_BadOperation (-27)	Parameters specify an action that is invalid.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.
Constants.FE_BadObjId (-3)	Invalid cell.

**Returns**

int

**Syntax**

StraddleCells(heightInRows, widthInCols)

**Parameters**

Parameter name	Data Type	Optional	Description
heightInRows	int	No	Denotes the number of cells to straddle vertically.
widthInCols	int	No	Indicates the number of cells to straddle horizontally.

**UnStraddleCells****Description**

The UnStraddleCells() method unstraddles the specified cells in a table.

The UnStraddleCells() method returns `FE_Success` on success. The method assigns one of the following values to `FA_errno`, on failure:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker product does not support tables.
Constants.FE_BadOperation (-27)	Parameters specify an action that is invalid.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.
Constants.FE_BadObjId (-3)	Invalid cell.

### Returns

int

### Syntax

UnStraddleCells(heightInRows, widthInCols)

### Parameters

Parameter name	Data Type	Optional	Description
heightInRows	int	No	The number of cells to unstraddle vertically.
widthInCols	int	No	The number of cells to unstraddle horizontally.

## GetText

### Description

The `GetText()` method gets the text from the following types of objects:

- Cell
- Element
- Flow
- Fn
- Pgf
- SubCol

- TextFrame
- TextLine
- TiApiClient
- TiFlow
- TiText
- TiTextTable
- XRef

Call this method on the document as follows:

```
document.GetText(flags)
```

`GetText()` returns an object containing an array of text items. Each text item contains either a string of text, an object that appears within the text (such as a table or an anchored frame), an indicator that the text properties have changed, or the object that organizes the text (such as a paragraph or a text column).

**NOTE:** Use the `DeallocateTextItems()` method to deallocate the memory used by the `TextItems` structure returned by `GetText()` when you have finished using it.

To get specific types of text items, OR the constants that represent them (for example, use a bitwise OR to combine `FTI_FlowBegin` and `FTI_String`) into flags. To get all types of text items, specify -1.

The method returns a `TextItems` object containing the array of text items of the requested text types.

`TextItem.dataType` can be one of the following constants:

Text Item Type	Text Item Represents	Text Item Data
<code>FTI_CharPropsChange</code>	A change in the text properties	Flags indicating which properties have changed (see the following table).
<code>FTI_ElementBegin</code>	The beginning of a container structural element	<code>FO_Element</code>
<code>FTI_ElementEnd</code>	The end of a container structural element	<code>FO_Element</code>
<code>FTI_ElemPrefixBegin</code>	The beginning of an element's prefix	<code>FO_Element</code>
<code>FTI_ElemPrefixEnd</code>	The end of an element's prefix	<code>FO_Element</code>
<code>FTI_ElemSuffixBegin</code>	The beginning of an element's suffix	<code>FO_Element</code>
<code>FTI_ElemSuffixEnd</code>	The end of an element's suffix	<code>FO_Element</code>
<code>FTI_FlowBegin</code>	The beginning of a flow	<code>FO_Flow</code>

<b>Text Item Type</b>	<b>Text Item Represents</b>	<b>Text Item Data</b>
FTI_FlowEnd	The end of a flow	FO_Flow
FTI_FnAnchor	A footnote	FO_Fn
FTI_FrameAnchor	An anchored frame	FO_AFrame
FTI_LineBegin	The beginning of a line	-
FTI_LineEnd	The end of a line and the line end type	If the line end is a normal line end, the value is 0; if it is a forced line end, the FTI_HardLineEnd flag is set; if it is a hyphen line end, the FTI_HyphenLineEnd flag is set
FTI_MarkerAnchor	A Marker	FO_Marker
FTI_PageBegin	The beginning of a page	FO_Page
FTI_PageEnd	The end of a page	FO_Page
FTI_PgfBegin	The beginning of a paragraph	FO_Pgf
FTI_PgfEnd	The end of a paragraph	FO_Pgf
FTI_String	A string of characters with the same condition and character format.	A character string
FTI_SubColBegin	The beginning of a column	FO_SubCol
FTI_SubColEnd	The end of a column	FO_SubCol
FTI_TblAnchor	A Table	FO_Tbl
FTI_TextFrameBegin	The beginning of a text frame	FO_TextFrame
FTI_TextFrameEnd	The end of a text frame	FO_TextFrame
FTI_TextInsetBegin	The beginning of a text inset	FO_TiApiClient, FO_TiFlow, FO_TiText, or FO_TiTextTable
FTI_TextInsetEnd	The end of a text inset	FO_TiApiClient, FO_TiFlow, FO_TiText, or FO_TiTextTable
FTI_TextObjId	The object to which the offsets of all the text items are relative	FO_Pgf or FO_TextLine
FTI_VarBegin	The beginning of a variable	FO_Var
FTI_VarEnd	The end of a variable	FO_Var

Text Item Type	Text Item Represents	Text Item Data
FTI_XRefBegin	The beginning of a cross-reference	FO_XRef
FTI_XRefEnd	The end of a cross-reference	FO_XRef

The following table lists the bit flags that a client can bitwise AND with the `idata` field of a `Constants.FTI_CharPropsChange` (0x00200000) text item. For example, to determine if the font family changed, bitwise AND the `Constants.FTF_FAMILY` (0x80000000) flag with the `idata` field.

Flags	Meaning
FTF_ALL	OR of all the flags
FTF_ANGLE	The font angle has changed.
FTF_CAPITALIZATION	The capitalization has changed.
FTF_CHANGEBAR	The change bars have changed.
FTF_CHARTAG	The Character Catalog format has changed.
FTF_COLOR	The color has changed.
FTF_CONDITIONTAG	The condition tag has changed.
FTF_ENCODING	The text encoding has changed,
FTF_FAMILY	The font family has changed.
FTF_IIF	An internal flag having to do with Asian text. input. If there is a non-zero value for this flag, a front-end processor is controlling that text and you should not modify the associated text item.
FTF_KERNX	The kern-x characteristic has changed.
FTF_KERNY	The kern-y characteristic has changed.
FTF_LANGUAGE	Character language has changed.
FTF_OUTLINE	The outline characteristic has changed.
FTF_OVERLINE	The overline characteristic has changed.
FTF_PAIRKERN	The pair kerning has changed.
FTF_POSITION	The character position has changed.
FTF_SHADOW	The shadow characteristic has changed.
FTF_SIZE	The font size has changed.

Flags	Meaning
FTF_SPREAD	The font spread has changed.
FTF_STRETCH	The font stretch value has changed.
FTF_STRIKETHROUGH	The strikethrough characteristic has changed.
FTF_TSUME	The Tsume setting has changed.
FTF_UNDERLINING	The underlining has changed.
FTF_VARIATION	The font variation has changed.
FTF_WEIGHT	The font weight has changed.

On success, the method returns a `TextItems` structure containing the array of text items of the requested types.

On failure, the method sets the `len` field of the returned `TextItems` structure to 0, and returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadObjId (-3)</code>	Invalid Object
<code>Constants.FE_NotTextObject (-41)</code>	Object specified for the text range is not an object that contains text.

If you call `GetText()` for an object (`FO_Element` object), the returned information depends on the type of element, as shown:

Value	Information Returned
<code>Constants.FV_FO_CONTAINE R (1)</code>	All the text items from the beginning to the end of the element.
<code>Constants.FV_FO_SYS_VAR (2)</code>	All the text items from the beginning to the end of the variable.
<code>Constants.FV_FO_XREF (3)</code>	All the text items from the beginning to the end of the cross reference.
<code>Constants.FV_FO_FOOTNOTE (5)</code>	All the text items from the beginning to the end of the footnote.
<code>Constants.FV_FO_TBL_TITLE (9)</code>	All the text items from the beginning to the end of the table title.

Value	Information Returned
Constants.FV_FO_TBL_CELL (14)	All the text items from the beginning to the end of the cell.
Constants.FV_FO_TBL_HEADING (10)	Nothing. GetText() failed.
Constants.FV_FO_TBL_BODY (11)	
Constants.FV_FO_TBL_FOOTING (12)	
Constants.FV_FO_MARKER (4)	
Constants.FV_FO_TBL (8)	
Constants.FV_FO_GRAPHIC (6)	
Constants.FV_FO_EQN (7)	
Constants.FV_FO_TBL_ROW (13)	

## Returns

*TextItems*

## Syntax

GetText (flags)

## Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

Retrieves the properties of the specified cell object.

See [GetProps](#) under the AFrame class for more information.

**Returns***PropVals***Syntax**`GetProps()`**SetProps****Description**

Sets the desired property of the cell object.

See [SetProps](#) under the AFrame class for more information.

**Returns**

Void

**Syntax**`SetProps(setVal)`**Parameters**

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The value of the property to set.

**ObjectValid****Description**

Indicates whether a cell object is valid. Returns 1 if the cell object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

**Returns**

int

**Syntax**`ObjectValid()`



# CharFmt

## Delete

### Description

Deletes the specified CharFmt object.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Get the properties of the specified CharFmt object.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the value of the property for the specified CharFmt object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The value of the property to set.

## ObjectValid

### Description

Indicates whether the specified CharFmt object is valid. Returns 1 if the CharFmt object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## CMSObject

### CMS SetProperty()

#### Description

Sets a single property for a CMS object.

#### Returns

void

#### Syntax

```
void CMS SetProperty (cmsSessionId, objectId, propval);
```

## Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
propVal	int	No	The specified property of a CMS object as a PropVal data structure

If CMS SetProperty() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_CMSRootObjectExists	The method tries to set a root which already exists.
Constants.FE_CMSBadItemFileType	The file type expected by the cms object does not match the valid file type.
Constants.FE_CMSBadItemType	The item type expected by the cms object does not match the valid item type
Constants.FE_CMSBadItemContainerType	The container value expected by the cms object is not properly set

## CMS SetProperty()

### Description

Sets multiple properties of a CMS object

### Returns

void

### Syntax

```
CMS SetProperty (cmsSessionId, objectId, propVals);
```

## Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
propVals	int	No	A property list that specifies the CMS properties . Properties are added as propvals with the identifier as CMSItemPropertyT enum and value.

If CMSSetProperties() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_CMSRootObjectExists	The method tries to set a root which already exists.
Constants.FE_CMSBadItemType	The file type expected by the cms object does not match the valid file type.
Constants.FE_CMSBadItemContainerType	The item type expected by the cms object does not match the valid item type
Constants.FE_CMSBadItemContainerType	The container value expected by the CMS object is not properly set

## CMSGetProperty()

### Description

Gets a specified property of a CMS object.

### Returns

*PropVals*

**Syntax**

```
CMSGetProperty (cmsSessionId, objectId,propertyId);
```

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session.
objectId	int	No	The ID of the CMS Object.
propertyId	int	No	F_PropldentT which allows user to specify property identifier as Integer value OR string value based on the CMS. For e.g. Documentum works on object name, whereas Generic CMS works on Integer based identifier ID of the CMS Object.

If CMSGetProperty() fails, the method assigns following values to FA\_erno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_BadParameter	The function call specified an invalid parameter.

**CMSGetProperty()****Description**

Gets the properties of a CMS object.

**Returns**

*PropVals*

**Syntax**

```
CMSGetProperty (cmsSessionId, objectId);
```

## Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
objectId	int	No	The ID of the CMS Object

If CMSGetProperties() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.

## CMSEnableCommand()

### Description

Enables the specified CMS command in the context menu of the CMS tree within FrameMaker.

### Returns

void

### Syntax

```
CMSEnableCommand (cmsSessionId, objectId, commandId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
commandId	int	No	The command to enable

If CMSEnableCommand() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_CMSBadCommandId	The client specified an invalid command ID.

## CMSTDisableCommand()

### Description

Disables the specified CMS command in the context menu of the CMS tree within FrameMaker.

### Returns

void

### Syntax

```
CMSTDisableCommand (cmsSessionId, objectId, commandId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
commandId	int	No	The command to disable

If CMSTDisableCommand() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.

Error	Reason
Constants.FE_CMSBadComm andId	The client specified an invalid command ID.

## CMSShowCheckoutUI()

### Description

Displays the checkout dialog for a CMS object

### Returns

bool

### Syntax

```
CMSShowCheckoutUI (sessionId, objectId, hideUiItems);
```

### Parameters

Parameter name	Data Type	Optional	Description
sessionId	int	No	The ID of the CMS session
objectId	int	No	The ID of the CMS Object
hideUiItems	int	No	Parameter to customize the checkout dialog. It can take the values from CMSCustomizeCheckoutUI enum

Enum constants used to customize CMS Object's Checkout user interface.

The possible values of the hideUiItems field are

Value for flags	Meaning
FV_CMSCheckoutUI_Id_ShowDependents	Flag to hide "Show dependents" checkbox

If CMSShowCheckoutUI() fails, the method assigns following values to FA\_erno:



Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSShowCheckinUI()

### Description

Displays the checkin dialog for a CMS object

### Returns

bool

### Syntax

```
CMSShowCheckinUI (sessionId, objectId, hideUiItems);
```

### Parameters

Parameter name	Data Type	Optional	Description
sessionId	int	No	The ID of the CMS session
objectId	int	No	The ID of the CMS Object
hideUiItems	int	No	Parameter to customize the checkout dialog. It can take the values from CMSCustomizeCheckinUI enum

Enum constants used to customize CMS Object's Checkin user interface.

The possible values of the hideUiItems field are

hideUiItems constants	Reason
Constants.FV_CMSCheckinUI_Id_MajorVersion	Flag to hide "Major version" radio button

hideUIitems constants	Reason
Constants.FV_CMSCheckinUI_Id_VersionLabel	Flag to hide "Version Label" text field
Constants.FV_CMSCheckinUI_Id_Description	Flag to hide "Description" text field
Constants.FV_CMSCheckinUI_Id_CheckinComment	Flag to hide "Checkin comment" text field
Constants.FV_CMSCheckinUI_Id_MakeThisCurrentVersion	Flag to hide "Make this current version" checkbox

If CMSShowCheckinUI() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSShowCancelCheckoutUI()

### Description

Displays the Cancel Check out dialog for a CMS object

### Returns

void

### Syntax

```
CMSShowCancelCheckoutUI (sessionId, objectId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session
cmsObjectId	int	No	The ID of the CMS object

If `CMSShowCancelCheckoutUI()` fails, the method assigns following values to `FA_erno`:

Error	Reason
<code>Constants.FE_CMSBadSessionId</code>	The client specified an invalid session ID
<code>Constants.FE_CMSBadObjectId</code>	The client specified an invalid cms object ID

## CMSShowDeleteUI()

### Description

Displays the Delete dialog for a CMS object

### Returns

`CMSShowDeleteParam`

### Syntax

```
CMSShowDeleteUI (F_ObjHandleT cmsSessionId, objectId, T hideUiItems);
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>sessionId</code>	<code>int</code>	No	The ID of the CMS session
<code>objectId</code>	<code>int</code>	No	The ID of the CMS Object
<code>hideUiItems</code>	<code>int</code>	No	Parameter to customize the delete dialog. It can take the values from <code>CMSCustomizeDeleteUI</code> enum

Enum constants used to customize CMS Object's Delete user interface.

The user can specify one or more of the following flag constants (using the OR expression for multiple flags) into the `hideUiItems` field:

Value for flags	Meaning
<code>Constants.FV_CMSDeleteUI_DeleteAllVersion</code>	Flag to hide "Delete all the version of a file" checkbox

Value for flags	Meaning
Constants.FV_CMSDeleteUI_DeleteAllDependents	Flag to hide “Delete all the dependents of a file” checkbox

If CMSShowDeleteUI() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSShowCommonListUI()

### Description

Displays the list-based dialogs such as “Show Version”, “Show Checked out files”, “Show dependents” and “Show result”. These items are fetched using the CMSGetItems CMS command.

### Returns

bool

### Syntax

```
CMSShowCommonListUI (cmsSessionId, objectId, commandId, title,
columnProperties);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
commandId	int	No	The ID of the CMS command
title	int	No	The Title of the Show Dialog
columnProperties	int	No	Properties Columns to show

If `CMSShowCommonListUI()` fails, the method assigns following values to `FA_errno`:

Error	Reason
<code>Constants.FE_CMSBadSessionId</code>	The client specified an invalid session ID.
<code>Constants.FE_CMSBadObjectId</code>	The client specified an invalid cms object ID.
<code>Constants.FE_CMSBadCommandId</code>	The client specified an invalid command ID.
<code>Constants.FE_BadParameter</code>	The function call specified an invalid parameter.

## CMSShowPropertyUI()

### Description

Displays the Property dialog for a CMS object

### Returns

`CMSProperties`

### Syntax

```
CMSShowPropertyUI ( cmsSessionId, objectId, props);
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>cmsSessionId</code>	<code>int</code>	No	The ID of the CMS Session
<code>cmsObjectId</code>	<code>int</code>	No	The ID of the CMS Object
<code>props</code>	<code>int</code>	No	<code>CMSPropertiesT</code> structure specifies a set of CMS object properties. Set NULL if user wants to show default properties

If `CMSShowPropertyUI()` fails, the method assigns following values to `FA_errno`:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSShowPropertyUIWithTitle

### Description

Displays the Property dialog for a CMS object with a specified title.

### Returns

CMSProperties

### Syntax

```
CMSShowPropertyUIWithTitle ( cmsSessionId, objectId, props, title);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
props	int	No	CMSPropertiesT structure specifies a set of CMS object properties. Set NULL if user wants to show default properties
title	string	No	Title of the dialog box

If CMSShowPropertyUIWithTitle fails, the method assigns following values to FA\_erno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID

Error	Reason
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSRegistration

### CMSGetCMSInfo()

#### Description

Gets the CMS information for a particular CMS registration id.

#### Returns

CMSInfo

#### Syntax

```
CMSGetCMSInfo (cmsId);
```

#### Parameters

Parameter name	Data Type	Optional	Description
cmsId	int	No	The registration ID of the CMS

If CMSGetCMSInfo() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadParameter	The function call specified an invalid parameter.

### CMSConfigLoginUI()

#### Description

Configures the CMS Connector Manager dialogs within the FrameMaker interface

#### Returns

void

## Syntax

```
CMSSessionLoginUI (cmsId, userFields, userLoginUI);
```

## Parameters

Parameter name	Data Type	Optional	Description
cmsId	int	No	The ID of the CMS
userFields	int	No	Optional user fields with strings. User can add upto two user fields.
userLoginUI	int	No	1 if user want to impement his own custom connection manager dialog

If CMSSessionLoginUI() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSSession

### CMSSessionLogout()

#### Description

Logs out the user from a particular CMS connection

#### Returns

void

#### Syntax

```
CMSSessionLogout(cmsSessionId);
```

#### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session



If CMSLogout() fails, the method assigns following values to FA\_erno:

Error	Reason
Constants.FE_CMSFailedLogout	Failed to log out of the CMS

## GetCMSObjectFromPath()

### Description

Gets CMS object from a URL path

### Returns

CMS Object

### Syntax

```
GetCMSObjectFromPath (cmsSessionId, urlPath)
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
urlPath	int	No	The url pathname of the file or folder

If GetCMSObjectFromPath() fails, the method assigns following values to FA\_erno:

Error	Reason
Constants.FE_CMSFailedGetItemFromPath	Failed to get the CMS object from the specified path

## CMSShowBrowseRepositoryUI

### Description

Displays repository browser dialog based on flag "showContainerOnly"

### Returns

CMS Object

**Syntax**

```
CMSShowBrowseRepositoryUI (cmsSessionId, showContainerOnly);
```

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session
showContainerOnly	int	No	1 if only container item is shown 0 if all items are shown

**CMSCreateObject****Description**

Creates a CMS object.

**Returns**

CMS Object

**Syntax**

```
CMSCreateObject (cmsSessionId);
```

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session

If CMSCreateObject() fails, the method assigns following values to FA\_erno:

Error	Reason
Constants.FE_CMSSessionBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSCreateObjectCreationFailed	Method failed to create a cms object.

## CMSSetCmsIdFromSession

### Description

Gets the CMS registration id from CMS Session id

### Returns

CMS Object

### Syntax

```
F_ApiCMSSetCmsIdFromSession (cmsSessionId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session

If CMSSetCmsIdFromSession() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSSetCmsIdFromSession	The client specified an invalid session ID

## Color

### Delete

#### Description

Deletes the specified color object.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete ()
```

## GetProps

### Description

Retrieves the properties of the specified color object.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the value of the property for the color object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The value of the property to be set.

## ObjectValid

### Description

Determines whether a color object is valid. Returns 1 if the Color object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

ObjectValid()

## CombinedFont

**CombinedFont****Description**

`CombinedFont` specifies a set of font characteristics for a combined font. The `combinedFont` field specifies a `CombinedFont` object. From this object you can get information about the combined font such as the base and Western font families, or the combined font name.

**Returns***CombinedFont***Syntax**`CombinedFont(combinedFont, variation, weight, angle)`**Parameters**

Parameter name	Data Type	Optional	Description
<code>combinedFont</code>	<i>CombinedFontDefn</i>	No	CombinedFont object.
<code>variation</code>	uint	No	Index of the font variation.
<code>weight</code>	uint	No	Index of the font weight.
<code>angle</code>	uint	No	Index of the font angle.

# CombinedFontDefn

## CombinedFamilyFonts

### Description

Returns the permutations of angles, variations, and weights available for a specified combined font definition.

The method does not take any arguments.

The `CombinedFamilyFonts()` method returns a `CombinedFonts` object that provides a list of the permutations of angles, variations, and weights available for the specified combined font definition.

### Returns

*CombinedFonts*

### Syntax

```
CombinedFamilyFonts()
```

## Delete

### Description

Deletes the specified `CombinedFonts` object.

See [Delete](#) under the `AFrame` class for more information.

### Returns

```
int
```

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the specified `CombinedFonts` object.

See [GetProps](#) under the `AFrame` class for more information.

### Returns

*PropVals*

## Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the value of the desired property of the CombinedFonts object.

See [SetProps](#) under the AFrame class for more information.

### Returns

```
Void
```

## Syntax

```
SetProps (setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The value of the property to be set.

## ObjectValid

### Description

Indicates whether a CombinedFonts object is valid. Returns 1 if the CombinedFonts object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

```
int
```

## Syntax

```
ObjectValid()
```

## CombinedFonts

An Array of *CombinedFont* objects with integer indexing and a *length* property.

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

*CombinedFonts*

#### Syntax

`concat (value)`

#### Parameters

Parameter name	Data Type	Optional	Description
value	<i>CombinedFonts</i>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

#### Returns

*CombinedFont*

#### Syntax

`pop ()`

### push

#### Description

Returns the new length of the array.



**Returns**

number

**Syntax**

`push(value)`

**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>Combine dFont</i>	No	The value to push into the array

## Command

**Delete****Description**

Deletes the specified Command object.

See [Delete](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

`Delete()`

**GetProps****Description**

Retrieves the properties of the specified Command object.

See [GetProps](#) under the AFrame class for more information.

**Returns**

*PropVals*

## Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the value of the specified Command property.

See [SetProps](#) under the AFrame class for more information.

### Returns

```
Void
```

## Syntax

```
SetProps (setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The value of the property to set.

## ObjectValid

### Description

Identifies whether a Command object is valid or not. Returns 1 if the Command object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

```
int
```

## Syntax

```
ObjectValid()
```

## Delete

### Description

Deletes the specified Command object.

The method does not take any arguments.

Call the Delete() method directly on the object to be deleted.

### Returns

int

### Syntax

```
Delete()
```

## CompareRet

### CompareRet

#### Description

CompareRet provides the results of a call to the Compare() method. The method returns a CompareRet object with the summary and composite documents.

#### Returns

*CompareRet*

#### Syntax

```
CompareRet (sum, comp)
```

#### Parameters

Parameter name	Data Type	Optional	Description
sum	<i>Doc</i>	No	Summary Document.
comp	<i>Doc</i>	No	Composite Document.

# CondFmt

## Delete

### Description

Deletes the specified `CondFmt` object.

See [Delete](#) under the `AFrame` class for more information.

### Returns

`int`

### Syntax

```
Delete()
```

## DeleteCondTag

### Description

Deletes a conditional tag from a document.

### Returns

`Status`

### Syntax

```
DeleteCondTag(docId, condTagId, action);
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>docId</code>	<code>int</code>	No	The id of the document containing the conditional tag to be deleted
<code>condTagId</code>	<code>int</code>	No	The id of the conditional tag object (type: <code>FO_CondFmt</code> ) in the doc
<code>action</code>	<code>int</code>	No	One of the following: <code>FF_UNTAGGED_ASK</code> : Prompt the user <code>FF_UNTAGGED_UNCOND</code> : Make text unconditional <code>FF_UNTAGGED_DELETE</code> : Delete text

If `DeleteCondTag()` fails, the method assigns following values to `FA_errno`:

Error	Reason
<code>Constants.FE_ReadOnly</code>	Document is read only.
<code>Constants.FE_BadValue</code>	Action is not one of the specified values
<code>Constants.FE_Success</code>	Deletion was successful

## GetProps

### Description

Retrieves the properties of the specified `CondFmt` object.  
See [GetProps](#) under the `AFrame` class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the value of the specified `CondFmt` object property.  
See [SetProps](#) under the `AFrame` class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The value of the property to set.

## ObjectValid

### Description

Identifies whether a `CondFmt` object is valid or not. Returns 1 if the `CondFmt` object is valid, else returns 0.

See [ObjectValid](#) under the `AFrame` class for more information.

### Returns

`int`

### Syntax

`ObjectValid()`

## Doc

### AddText

#### Description

Inserts text into a paragraph or a text line.

The text you specify for text must use the FrameMaker character set. To add special characters, you must specify octal (`\`) or hexadecimal (`\x`) sequences. The following table lists some of these sequences.

Special Character	Hexadecimal Representation	Octal Representation
>	<code>\x3e</code>	<code>\76</code>
" (straight double quotation mark)	<code>\x22</code>	<code>\42</code>
" (left double quotation mark)	<code>\xd2</code>	<code>\322</code>

Special Character	Hexadecimal Representation	Octal Representation
" (right double quotation mark)	\xd3	323

The method returns the location of the text that was added, on success. On failure, it sets one of the following error numbers to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object.
Constants.FE_NotTextObject (-41)	The object that textLoc specifies is not a paragraph (FO_Pgf) or a text line (FO_TextLine).
Constants.FE_OffsetNotFound (-21)	The offset specified for the text location could not be found in the specified text object.
Constants.FE_ReadOnly (-6)	The document is read only.
Constants.FE_BadSelectionForOperation (-59)	The location that textLoc specifies is invalid. For example, it is inside a variable or outside the highest level element in a structured FrameMaker document.

## Returns

*TextLoc*

## Syntax

```
AddText(textLoc, text)
```

## Parameters

Parameter name	Data Type	Optional	Description
textLoc	<i>TextLoc</i>	No	The text location at which to add the text.
text	string	No	The text to add.

## CenterOnText

### Description

Centers a range of text such that the middle of the text appears in the middle of the document window.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadRange (-36)	The specified text range is invalid.
Constants.FE_NotTextObject (-41)	The object that <code>textRange</code> specifies is not a paragraph ( <code>FO_Pgf</code> ) or a text line ( <code>FO_TextLine</code> ).
Constants.FE_OffsetNotFound (-21)	The offset specified for the text location could not be found in the specified paragraph or text line.

### Returns

`int`

### Syntax

`CenterOnText (textRange)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>textRange</code>	<i>TextRange</i>	No	The range of text to center.

### Clear

#### Description

Deletes the current selection from a document.

If you specify 0 for flags, the method suppresses any interactive alerts or warnings that arise, leaves the selected table cells empty, and deletes hidden text.

You can OR the following values into flags:

Flags	Meaning
Constants.FF_INTERACTIVE (0x0001)	Prompt the user with dialog or alert boxes that arise.
Constants.FF_CUT_TBL_CELLS (0x0002)	Remove cleared table cells.



Flags	Meaning
Constants.FF_VISIBLE_ONLY (0x0010)	Clear only the visible portion of the selection.
Constants.FF_DONT_DELETE_HIDDEN_TEXT (0x0004)	Do not delete hidden text.

The `Constants.FF_INTERACTIVE` flag takes precedence over other flags. If you specify `Constants.FF_INTERACTIVE | Constants.FF_DONT_DELETE_HIDDEN_TEXT` and the selection contains hidden text, then FrameMaker allows you to select whether to delete the hidden text or to retain it.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current selection is invalid for this operation.
Constants.FE_Canceled (-44)	The User or parameters canceled the operation.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation.

## Returns

`int`

## Syntax

`Clear(flags)`

## Parameters

Parameter name	Data Type	Optional	Description
<code>flags</code>	<code>int</code>	No	Bit field that specifies how to clear the text and how to handle interactive alerts. For default settings, specify 0.

## ClearAllChangebars

### Description

Clears all change bars from a specified document.

*NOTE: The method performs the same function as clicking the Clear All Change Bars box in the Change Bars dialog box.*

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_WrongProduct (-60)</code>	The current FrameMaker version does not support the requested operation.
<code>Constants.FE_SystemError (-42)</code>	A system error occurred.

### Returns

`int`

### Syntax

```
ClearAllChangebars ()
```

## Compare

### Description

Compares two documents or two books.

You can OR the values shown in the following table into the flags argument.

Flags	Meaning
<code>Constants.FF_CMP_SUMMARY_ONLY (0x01)</code>	Generate a summary document, but not a composite document.
<code>Constants.FF_CMP_CHANGE_BARS (0x02)</code>	Turn on change bars in the composite document.
<code>Constants.FF_CMP_HYPERLINKS (0x04)</code>	Put hypertext links in the summary document.

Flags	Meaning
Constants.FF_CMP_SUMKIT (0x08)	Open the summary document.
Constants.FF_CMP_COMPKIT (0x10)	Open the composite document.

On success, the method returns a `CompareRet` structure containing the summary and composite documents detailing the changes found, if any.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadCompare (-35)	Documents to be compared are not of the same type.
Constants.FE_CompareTypes (-34)	One of the files is not a FrameMaker document or book, or one file is a book and the other is a document.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation.

## Returns

*CompareRet*

## Syntax

```
Compare(newerDoc, flags, insertCondTag, deleteCondTag, replaceText,
compareThreshold)
```

## Parameters

Parameter name	Data Type	Optional	Description
<code>newerDoc</code>	<i>Doc</i>	No	The newer version of the document to be compared.
<code>flags</code>	<code>int</code>	No	Bit flags that specify how to generate the summary and composite documents. Specify 0 for the default flags.

Parameter name	Data Type	Optional	Description
insertCondTag	string	No	The condition tag to apply to insertions shown in the composite document. For no insert condition tag, specify <code>null</code> .
deleteCondTag	string	No	The condition tag to apply to deletions shown in the composite document. For no delete condition tag, specify <code>null</code> .
replaceText	string	No	Text to appear in place of the deleted text. For no replacement text, specify <code>null</code> .
compareThreshold	int	No	Threshold expresses the percentage of words that can change before paragraphs are considered not equal. If two paragraphs are equal, word differences between them are shown within a paragraph in the composite document. If this value is set to zero, the compare function uses the default value of 75. If not zero, the function uses the setting in the FrameMaker configuration files.

## Copy

### Description

Copies the current selection to the FrameMaker Clipboard.

If the value of the flags parameter is 0, the method suppresses any interactive alerts or warnings that arise.

You can also OR the following values into flags:

Flags	Meaning
<code>Constants.FF_INTERACTIVE (0x0001)</code>	Prompt user with dialog or alert boxes that arise.
<code>Constants.FF_STRIP_HYPertext (0x0080)</code>	Do not copy any hypertext markers in the selection.
<code>Constants.FF_VISIBLE_ONLY (0x0010)</code>	Copy only the visible portion of the selection.

The `Constants.FF_INTERACTIVE (0x0001)` flag takes precedence over other flags. If you specify `Constants.FF_INTERACTIVE (0x0001) | Constants.FF_VISIBLE_ONLY (0x0010)` and the selection is not visible, then FrameMaker allows the user to choose whether to copy the selection or not.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_WrongProduct (-60)</code>	The current FrameMaker version does not support the requested operation
<code>Constants.FE_BadSelectionForOperation (-59)</code>	The Selection doesn't support the requested operation.
<code>Constants.FE_Canceled (-44)</code>	The User or parameters canceled the operation.
<code>Constants.FE_BadOperation (-27)</code>	The parameters specified an invalid operation.

### Returns

`int`

### Syntax

`Copy(flags)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>flags</code>	<code>int</code>	No	Bit field that specifies how to copy the text and how to handle interactive alerts. For default settings, specify 0.

### Cut

#### Description

Cuts the current selection to the FrameMaker Clipboard.

If the value of the `flags` parameter is 0, the method suppresses any interactive alerts or warnings that arise.

You can OR the following values into `flags`:

Flags	Meaning
<code>Constants.FF_INTERACTIVE (0x0001)</code>	Prompt user with dialog or alert boxes that arise.

Flags	Meaning
Constants.FF_CUT_TBL_CELLS (0x0002)	Remove cut table cells.
Constants.FF_VISIBLE_ONLY (0x0010)	Cut only the visible portion of the selection.
Constants.FF_DONT_DELETE_HIDDEN_T EXT (0x0004)	Do not cut hidden text.

The `Constants.FF_INTERACTIVE (0x0001)` flag takes precedence over other flags. If you specify `Constants.FF_INTERACTIVE (0x0001) | Constants.FF_DONT_DELETE_HIDDEN_TEXT (0x0004)` and the selection contains hidden text, then FrameMaker allows the user to choose whether to delete the hidden text or retain it.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation
Constants.FE_BadSelectionForOperation (-59)	The Selection doesn't support the requested operation.
Constants.FE_Canceled (-44)	The User or parameters canceled the operation.
Constants.FE_BadOperation (-27)	The parameters specified an invalid operation.

### Returns

`int`

### Syntax

`Cut (flags)`

### Parameters

Parameter name	Data Type	Optional	Description
flags	<code>int</code>	No	Bit field that specifies how to cut the text and how to handle interactive alerts. For default settings, specify 0.

## DeleteText

### Description

Deletes a specified text range from a document.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDelete (-17)	Specified text could not be deleted.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_NotTextObject (-41)	Object specified for the text range is not an object that contains text, for example, a text frame ( <code>FO_TextFrame</code> ), a paragraph ( <code>FO_Pgf</code> ) or a text line ( <code>FO_TextLine</code> ).
Constants.FE_BadSelectionForOperation (-59)	Selection is within a locked text range.

### Returns

`int`

### Syntax

`DeleteText (textRange)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>textRange</code>	<i>TextRange</i>	No	The text range to delete.

## DemoteElement

### Description

Demotes the selected structural element or elements. The element becomes a child of the sibling element before it.

**NOTE:** At least one structural element must be selected in the document, before calling `DemoteElement()`.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.

### Returns

Void

### Syntax

```
DemoteElement()
```

### GetTextForRange

#### Description

Gets the text for a specified text range.

Call the method on the document as follows:

```
document.GetTextForRange()
```

You can OR the following constants into the flags parameter, to retrieve various text items:

Text Item Type	Text Item Represents	Text Item Data
Constants.FTI_CharPropsChange (0x00200000)	A change in the text properties	Flags indicating which properties have changed (see the following table).
Constants.FTI_ElementBegin (0x00080000)	The beginning of a container structural element	FO_Element
Constants.FTI_ElementEnd (0x00100000)	The end of a container structural element	FO_Element
Constants.FTI_ElementPrefixBegin (0x08000000)	The beginning of an element's prefix	FO_Element



Text Item Type	Text Item Represents	Text Item Data
Constants.FTI_ElemPrefixEnd (0x10000000)	The end of an element's prefix	FO_Element
Constants.FTI_ElemSuffixBegin (0x20000000)	The beginning of an element's suffix	FO_Element
Constants.FTI_ElemSuffixEnd (0x40000000)	The end of an element's suffix	FO_Element
Constants.FTI_FlowBegin (0x00000020)	The beginning of a flow	FO_Flow
Constants.FTI_FlowEnd (0x00000040)	The end of a flow	FO_Flow
Constants.FTI_FnAnchor (0x00001000)	A footnote	FO_Fn
Constants.FTI_FrameAnchor (0x00000800)	An anchored frame	FO_AFrame
Constants.FTI_LineBegin (0x00000002)	The beginning of a line	-
Constants.FTI_LineEnd (0x00000004)	The end of a line and the line end type	If the line end is a normal line end, the value is 0; if it is a forced line end, the FTI_HardLineEnd flag is set; if it is a hyphen line end, the FTI_HyphenLineEnd flag is set
Constants.FTI_MarkerAnchor (0x00004000)	A Marker	FO_Marker
Constants.FTI_PageBegin (0x00000080)	The beginning of a page	FO_Page
Constants.FTI_PageEnd (0x00000100)	The end of a page	FO_Page
Constants.FTI_PgfBegin (0x00000008)	The beginning of a paragraph	FO_Pgf
Constants.FTI_PgfEnd (0x00000010)	The end of a paragraph	FO_Pgf
Constants.FTI_String (0x00000001)	A string of characters with the same condition and character format.	A character string

Text Item Type	Text Item Represents	Text Item Data
Constants.FTI_SubColBegin (0x00000200)	The beginning of a column	FO_SubCol
Constants.FTI_SubColEnd (0x00000400)	The end of a column	FO_SubCol
Constants.FTI_TblAnchor (0x00002000)	A Table	FO_Tbl
Constants.FTI_TextFrameBegin (0x00400000)	The beginning of a text frame	FO_TextFrame
Constants.FTI_TextFrameEnd (0x00800000)	The end of a text frame	FO_TextFrame
Constants.FTI_TextInsetBegin (0x02000000)	The beginning of a text inset	FO_TiApiClient, FO_TiFlow, FO_TiText, or FO_TiTextTable
Constants.FTI_TextInsetEnd (0x04000000)	The end of a text inset	FO_TiApiClient, FO_TiFlow, FO_TiText, or FO_TiTextTable
Constants.FTI_TextObject (0x01000000)	The object to which the offsets of all the text items are relative	FO_Pgf or FO_TextLine
Constants.FTI_VarBegin (0x00020000)	The beginning of a variable	FO_Var
Constants.FTI_VarEnd (0x00040000)	The end of a variable	FO_Var
Constants.FTI_XRefBegin (0x00008000)	The beginning of a cross-reference	FO_XRef
Constants.FTI_XRefEnd (0x00010000)	The end of a cross-reference	FO_XRef

The following table lists the flags for `FTI_CharPropsChange`. These flags indicate which properties have changed.

Flags	Meaning
Constants.FTF_ALL (0xFFFFE03)	OR of all the flags
Constants.FTF_ANGLE (0x10000000)	The font angle has changed.

Flags	Meaning
Constants.FTF_CAPITALIZATION (0x000040000)	The capitalization has changed.
Constants.FTF_CHANGE_BARS (0x01000000)	The change bars have changed.
Constants.FTF_CHARACTER_CATALOG (0x00008000)	The Character Catalog format has changed.
Constants.FTF_COLOR (0x00010000)	The color has changed.
Constants.FTF_CONDITION_TAG (0x00000001)	The condition tag has changed.
Constants.FTF_ENCODING (0x00000200)	The text encoding has changed,
Constants.FTF_FAMILY (0x80000000)	The font family has changed.
Constants.FTF_IIF (0x00000002)	An internal flag having to do with Asian text. input. If there is a non-zero value for this flag, a front-end processor is controlling that text and you should not modify the associated text item.
Constants.FTF_KERN_X (0x00080000)	The kern-x characteristic has changed.
Constants.FTF_KERN_Y (0x00040000)	The kern-x characteristic has changed.
Constants.FTF_LANGUAGE (0x00000800)	Character language has changed.
Constants.FTF_OUTLINE (0x00800000)	The outline characteristic has changed.
Constants.FTF_OVERLINE (0x02000000)	The overline characteristic has changed.
Constants.FTF_PAIR_KERNING (0x00200000)	The pair kerning has changed.
Constants.FTF_POSITION (0x00002000)	The character position has changed.
Constants.FTF_SHADOW (0x00400000)	The shadow characteristic has changed.

Flags	Meaning
Constants.FTF_SIZE (0x00100000)	The font size has changed.
Constants.FTF_SPREAD (0x00020000)	The font spread has changed.
Constants.FTF_STRETC H (0x00001000)	The font stretch value has changed.
Constants.FTF_STRIKET HROUGH (0x04000000)	The strikethrough characteristic has changed.
Constants.FTF_TSUME (0x00000400)	The Tsume setting has changed.
Constants.FTF_UNDER LINING (0x08000000)	The underlining has changed.
Constants.FTF_VARIATI ON (0x40000000)	The font variation has changed.
Constants.FTF_WEIGH T (0x20000000)	The font weight has changed.

On success, the method returns a `TextItems` structure containing the array of text items of the requested types.

On failure, the method sets the `len` field of the returned `TextItems` structure to 0, and returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_NotTextObject (-41)	Object specified for the text range is not an object that contains text.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_OffsetNotFoun d (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.

**NOTE:** The returned `TextItems` structure references memory that is allocated by the method. Use the `DeallocateTextItems()` method to free this memory when you are done with using it.

## Returns

*TextItems*

## Syntax

```
GetTextForRange(tr, flags)
```

## Parameters

Parameter name	Data Type	Optional	Description
tr	<i>TextRange</i>	No	The text range containing the text you want to get.
flags	int	No	Bit flags that specify the type of text items to retrieve. To get specific types of text items, OR the constants that represent them (for example, FTI_FlowBegin and FTI_String) into flags. To get all types of text items, specify -1.

## GetTextForRange2

### Description

Gets the text for a specified text range.

You can OR the same flags as in the `GetTextForRange()` method, for the flags parameter.

On success, the method returns a `TextItems` structure containing the array of text items of the requested types.

On failure, the method sets the `len` field of the returned `TextItems` structure to 0, and returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_NotTextObject (-41)	Object specified for the text range is not an object that contains text.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.

**NOTE:** The returned `TextItems` structure references memory that is allocated by the method. Use the `DeallocateTextItems()` method to free this memory when you are done with using it.

## Returns

`TextItems`

## Syntax

```
GetTextForRange2(tr, flags, flags2)
```

## Parameters

Parameter name	Data Type	Optional	Description
tr	<code>TextRange</code>	No	The text range containing the text you want to get.
flags	int	No	Bit flags for the base set text item types that of text items to retrieve.
flags2	int	No	Bit flags for the extended set of text item types that specify the type of text items to retrieve. To get specific types of text items from the extended set, OR the constants that represent them (for example, use a bitwise OR to combine <code>FTI2_RubiTextBegin</code> and <code>FTI2_RubiTextEnd</code> ) into flags2. To get all types of the extended set of text items, specify -1.

## GetTextProps

### Description

Gets the text properties (such as the format tag, font family and size, and conditions) for a location in text.

As the text properties can be different for each character, you can only get the text properties for an individual location in the text.

On success, the method returns a `PropVals` structure with the text property list for the specified location.

On failure, the method sets the `len` field of the returned `PropVals` structure to 0, and returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_NotTextObject (-41)	Object specified for the text range is not an object that contains text.
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation

**Returns***PropVals***Syntax**

GetTextProps (textLoc)

**Parameters**

Parameter name	Data Type	Optional	Description
textLoc	<i>TextLoc</i>	No	The text location of the character that you want to get text properties for. The returned properties are the properties that apply to the character to the right of the specified location.

**GetTextPropVal****Description**

Gets a text property (such as the format tag, font family and size, or conditions) for a location in text.

As a text property can be different for each character, you can get the property for only one location in text at a time.

On success, the method returns a *PropVal* structure with the details of the specified property. On failure, the method returns one of the following values to *FA\_errno*:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_BadPropNum (-4)	Specified property number is invalid.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation

### Returns

*PropVal*

### Syntax

```
GetTextPropVal(textLoc, propNum)
```

### Parameters

Parameter name	Data Type	Optional	Description
textLoc	<i>TextLoc</i>	No	The text location of the character for which you want to get the the text property. The returned property applies to the character to the right of this location.
propNum	int	No	The property to query. Specify a constant, such as <code>FontFamily</code> .

**NOTE:** The returned `PropVal` structure references memory that is allocated by the method. Use the `DeallocatePropVal()` method to free this memory when you are done with using it.

## GetTextVal

### Description

Gets the value of a specified text property, which can be of any type.



On success, the method returns a `TypedVal` structure containing the value of the specified property. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_BadPropNum (-4)	Specified property number is invalid.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_NotTextObject (-41)	Object specified for the text location is not an object that contains text.
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation

## Returns

*TypedVal*

## Syntax

```
GetTextVal(textLoc, propNum)
```

## Parameters

Parameter name	Data Type	Optional	Description
textLoc	<i>TextLoc</i>	No	The text location of the character for which you want to get the text property. The returned property applies to the character to the right of this location.
propNum	int	No	The property to query. Specify a constant, such as <code>FontFamily</code> .

## HypertextCommand

### Description

Simulates a user-invoked hypertext command.

**NOTE:** Calling the `HypertextCommand()` method has the same effect as a user clicking on a hypertext marker containing the specified text.

**IMPORTANT:** Ensure that you do not include any hypertext command in your custom menus. Because the hypertext commands can only be executed in the View Only mode, and in the View Only mode, the custom menus are not visible.

The method returns `FE_Success` on success.

On failure, the method returns the following value to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.

### Returns

int

### Syntax

`HypertextCommand(hypertext)`

### Parameters

Parameter name	Data Type	Optional	Description
hypertext	string	No	A hypertext command to execute, such as <code>gotolink</code> or <code>previouslink</code> . You can specify any command that is valid in a hypertext marker in the document.

## MergeIntoFirst

### Description

Merges the selected structural elements into the first element in the selection.

**NOTE:** At least two structural elements must be selected in the document before calling the `MergeIntoFirst()` method.

On success, the method does not return any value.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.

### Returns

Void

### Syntax

```
MergeIntoFirst()
```

## MergeIntoLast

### Description

Merges the selected structural elements into the last element in the selection.

**NOTE:** *At least two structural elements must be selected in the document before calling the MergeIntoLast() method.*

On success, the method does not return any value.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.

### Returns

Void

## Syntax

`MergeIntoLast()`

## NewTable

### Description

Inserts a table (`FO_Tbl` object).

When you create a table in the user interface, you can specify a Table Catalog format for the table. FrameMaker uses the following properties of the Table Catalog format as the defaults for the new table:

- Number of body rows (`TblInitNumBodyRows`)
- Number of columns (`TblInitNumCols`)
- Number of footer rows (`TblInitNumFRows`)
- Number of header rows (`TblInitNumHRows`)
- Paragraph formats for header, body, and footer cells

For example, if the Table Catalog format's `TblInitNumCols` property is set to 8, the `NumCols` property of the new table is set to 8. With the `NewTable()` method, you can use the Table Catalog format properties as defaults for the number of rows and columns in a new table, or you can provide your own defaults. After you have create a table, you can add or remove rows with the `AddRows()` and `DeleteRows()` methods. You can add or remove columns with the `AddCols()` and `DeleteCols()` methods. If you use the `NewTable()` method to create a table in a structured FrameMaker document, FrameMaker applies default element tags, such as `Table`, `Row`, and `Cell`, to the table element and its child elements. To make these elements valid, you must add code to change their tags. In most cases it is easier to add tables to structured documents by calling the `NewElementInHierarchy()` or `NewElement()` methods to add a table element.

On success, the method returns the newly added table.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadObjId (-3)</code>	Invalid Object
<code>Constants.FE_NotTextObject (-41)</code>	Object specified for the text location is not a paragraph ( <code>FO_Pgf</code> ).
<code>Constants.FE_OffsetNotFound (-21)</code>	Offset specified for the text location could not be found in the specified paragraph or text line.
<code>Constants.FE_BadOperation (-27)</code>	Function call specified an illegal operation.

Error	Reason
Constants.FE_BadNew (-23)	Table cannot be created; the format specified by <code>format</code> may not exist or the text location specified by <code>textLoc</code> is in a table or a footnote.

## Returns

*Tbl*

## Syntax

```
NewTable(format, numCols, numBodyRows, numHeaderRows, numFooterRows, textLoc)
```

## Parameters

Parameter name	Data Type	Optional	Description
<code>format</code>	<code>string</code>	No	The table format tag (for example, <code>FormatA</code> or <code>Wide Table</code> ). To use the default format, specify <code>NULL</code> .
<code>numCols</code>	<code>int</code>	No	The number of columns in the table. To use the default number of columns from the Table Catalog format, specify <code>-1</code> .
<code>numBodyRows</code>	<code>int</code>	No	The number of rows in the table. To use the default number of body rows from the Table Catalog format, specify <code>-1</code> .
<code>numHeaderRows</code>	<code>int</code>	No	The number of heading rows in the table. To use the default number of header rows from the Table Catalog format, specify <code>-1</code> .
<code>numFooterRows</code>	<code>int</code>	No	The number of footing rows in the table. To use the default number of footer rows from the Table Catalog format, specify <code>-1</code> .
<code>textLoc</code>	<i>TextLoc</i>	No	The location at which to insert the new table. The location cannot be within a footnote or a table.

## Paste

### Description

Pastes the contents of the FrameMaker Clipboard into a specified document at the insertion point. Cutting and Pasting objects causes FrameMaker to create an ID for the pasted object.

Specifying a value of 0 for the flags parameter, causes the `Paste()` method to suppress any interactive alerts or warnings that arise. In addition, the method inserts columns to the left of the current columns and rows above the current row.

You can OR the following values into flags:

Flags	Meaning
<code>Constants.FF_INTERACTIVE (0x0001)</code>	Prompt the user with dialog or alert boxes that arise.
<code>Constants.FF_VISIBLE_ONLY (0x0010)</code>	Cut only the visible portion of the selection.
<code>Constants.FF_DONT_DELETE_HIDDEN_TEXT (0x0004)</code>	Do not replace hidden text.
<code>Constants.FF_DONT_APPLY_ALL_ROWS (0x0040)</code>	Do no't apply condition setting on the Clipboard to all rows. If whole table is selected and the Clipboard contains condition setting, cancel the paste operation.
<code>Constants.FF_REPLACE_CELLS (0x0020)</code>	Replace selected cells with cells on the Clipboard.
<code>Constants.FF_INSERT_BELOW_RIGHT (0x0008)</code>	Add columns to the right of the current column or below the current row.

When you use the `Paste()` method to paste table cells into a table, it does not work exactly like the interactive Paste command.

The interactive Paste command automatically overwrites cells if the Clipboard contains less than an entire row or column. For example, if the insertion point is in a three-column table and the Clipboard contains a single cell, the interactive Paste command overwrites the cell containing the insertion point with the cell on the Clipboard.

If two cells in the table are selected, the interactive Paste command overwrites both of them with the cell on the Clipboard.

By default, the `Paste()` method does not overwrite any cells. If the Clipboard contains less than an entire row or column when you call `Paste()`, or if the current selection is less than an entire row, the `Paste()` method does nothing and returns `Constants.FE_BadSelectionForOperation (-59)`. The method ensures that you do not inadvertently overwrite any cells.

To make the `Paste()` method replace cells with the Clipboard contents, you must call it with the `Constants.FF_REPLACE_CELLS (0x0020)` flag set. The `Constants.FF_INTERACTIVE (0x0001)` flag takes precedence over other flags.

If you specify `Constants.FF_INTERACTIVE (0x0001) | Constants.FF_DONT_DELETE_HIDDEN_TEXT (0x0004)` and the selection contains hidden text, FrameMaker prompts the user, and allows the user to choose whether to delete the hidden text or retain it.

It is illegal to specify `Constants.FF_REPLACE_CELLS (0x0020) | Constants.FF_INSERT_BELOW_RIGHT (0x0008)`.

The method returns `FE_Success` on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadOperation (-27)</code>	Function call specified an illegal operation.
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadSelectionForOperation (-59)</code>	Current text selection is invalid for this operation.
<code>Constants.FE_Canceled (-44)</code>	User canceled the operation.

## Returns

`int`

## Syntax

`Paste(flags)`

## Parameters

Parameter name	Data Type	Optional	Description
<code>flags</code>	<code>int</code>	No	Bit field that specifies how to paste the text and how to handle interactive alerts. For default settings, specify 0.

## PromoteElement

### Description

Promotes the selected structural element. The selected element becomes a sibling of its former parent and appears immediately after its former parent. The siblings that follow it become the element's children.

**NOTE:** One structural element must be selected when the `PromoteElement()` method is called. The selected element cannot be the root element or a child of the root element.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	The current FrameMaker version does not support the requested operation
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadSelectionForOperation (-59)</code>	Current text selection is invalid for this operation.

### Returns

Void

### Syntax

```
PromoteElement()
```

## QuickSelect

### Description

Implements a quick-key interface that allows the user to choose a string from a list of strings in the document Tag area.

The `QuickSelect()` method highlights the document Tag area and displays a prompt and the first string in a specified list of strings.

The user can display a string in the Tag area by typing the first few letters of the string. The user can also scroll through the strings by pressing the up and down arrow keys.

To choose a string, the user presses Return when the string appears in the Tag area. To cancel the choice, the user clicks in the document without pressing Return.

On success, the method returns an index into the array of strings specified by `stringlist` or `-1` if the user cancels the quick selection.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_Transport (-1)</code>	A transport error occurred.
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.



**Returns**

int

**Syntax**

QuickSelect(prompt, stringlist)

**Parameters**

Parameter name	Data Type	Optional	Description
prompt	string	No	The prompt that appears in the Tag area
stringlist	<i>Strings</i>	No	The list of strings from which the user can choose

**Redisplay****Description**

Updates the display for a specified document to reflect any changes that occurred when the `Displaying` property was set to `False`. If you have set the `Displaying` property to `False` and subsequently reset it to `True`, you should call the `Redisplay()` method to redisplay each document you modified.

The method returns `FE_Success` on success. On failure, the method returns the following value to `FA_erno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.

**Returns**

int

**Syntax**

Redisplay()

## Reformat

### Description

Reformats the specified document. If you have disabled and subsequently re-enabled reformatting by setting the session property, `Reformatting` to `True`, you should call the `Reformat()` method to reformat each changed document in the session.

The method returns `FE_Success` on success. On failure, the method returns the following value to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.

### Returns

`int`

### Syntax

`Reformat()`

## Rehyphenate

### Description

Rehyphenates a specified document based on changes the user has made to words' hyphenation points.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker version does not support this operation.
<code>Constants.FE_SystemError (-42)</code>	Could not allocate memory.

### Returns

`int`

## Syntax

Rehyphenate ()

## ResetEqnSettings

### Description

Resets the document equation settings to the default settings.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.
Constants.FE_SystemError (-42)	Could not allocate memory.

### Returns

int

### Syntax

ResetEqnSettings ()

## ResetReferenceFrames

### Description

Resets the reference frames in the specified document. This method is useful for updating a document after you have programmatically changed a reference frame that is referenced by paragraphs in the document.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.

Error	Reason
Constants.FE_SystemError (-42)	Could not allocate memory.

**Returns**

int

**Syntax**

```
ResetReferenceFrames()
```

**RestartPgfnNumbering****Description**

Restarts the paragraph numbering for a specified document.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.
Constants.FE_SystemError (-42)	Could not allocate memory.

**Returns**

int

**Syntax**

```
RestartPgfnNumbering()
```

**ScrollToText****Description**

Scrolls the document window to a specified text range. The method scrolls the document to the end of the range that is closest to the current display position.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_NotTextObject (-41)	Object specified for the text location is not a paragraph (FO_Pgf) or a flow (FO_Flow).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text range could not be found in the specified paragraph or text line.
Constants.FE_BadRange (-36)	The specified text range is invalid.

**Returns**

int

**Syntax**

ScrollToText (textRange)

**Parameters**

Parameter name	Data Type	Optional	Description
textRange	<i>TextRange</i>	No	The text range to which to scroll.

**SetTextProps****Description**

Sets the text properties (such as the format tag, font family, and size) for a text range.

On success, the method does not return anything.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadName (-33)	Specified name is illegal.
Constants.FE_BadRange (-36)	Specified text range is invalid.

Error	Reason
Constants.FE_GenRuleAmbiguous (-55)	General rule in structured document is ambiguous.
Constants.FE_GenRuleConnectorExpected (-57)	General rule in structured document is missing a connector.
Constants.FE_GenRuleItemExpected (-51)	General rule in structured document is missing a rule item.
Constants.FE_GenRuleLeftBracketExpected (-53)	General rule in structured document is missing a left bracket.
Constants.FE_GenRuleMixedConnectors (-52)	General rule in structured document has mixed connectors.
Constants.FE_GenRuleRightBracketExpected (-54)	General rule in structured document is missing a right bracket.
Constants.FE_GenRuleSyntaxError(-56)	General rule in structured document has a syntax error.
Constants.FE_NotTextObject (-41)	Object specified for the text range is not a paragraph (FO_Pgf) or a flow (FO_Flow).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text range could not be found in the specified paragraph or text line.
Constants.FE_OutOfRange (-7)	Specified property value is out of the legal range for the specified property.
Constants.FE_ReadOnly (-6)	Property is read-only and cannot be set.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.

**Returns**

Void

**Syntax**

SetTextProps (textRange, setVal)

## Parameters

Parameter name	Data Type	Optional	Description
textRange	<i>TextRange</i>	No	The text range to which to apply the property list.
setVal	<i>PropVals</i>	No	The property list to apply to the text range.

## SetTextPropVal

### Description

Sets a text property for a specified text range. The property can be of any type.

On success, the method does not return anything.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadName (-33)	Specified name is illegal.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_BadPropNum (-4)	Specified property number is invalid.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_GenRuleAmbiguous (-55)	General rule in structured document is ambiguous.
Constants.FE_GenRuleConnectorExpected (-57)	General rule in structured document is missing a connector.
Constants.FE_GenRuleItemExpected (-51)	General rule in structured document is missing a rule item.
Constants.FE_GenRuleLeftBracketExpected (-53)	General rule in structured document is missing a left bracket.
Constants.FE_GenRuleMixedConnectors (-52)	General rule in structured document has mixed connectors.
Constants.FE_GenRuleRightBracketExpected (-54)	General rule in structured document is missing a right bracket.

Error	Reason
Constants.FE_GenRuleSyntaxError(-56)	General rule in structured document has a syntax error.
Constants.FE_NotTextObject (-41)	Object specified for the text range is not a paragraph (FO_Pgf) or a flow (FO_Flow).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text range could not be found in the specified paragraph or text line.
Constants.FE_OutOfRange (-7)	Specified property value is out of the legal range for the specified property.
Constants.FE_ReadOnly (-6)	Property is read-only and cannot be set.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.

### Returns

Void

### Syntax

```
SetTextPropVal(textRange, setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
textRange	<i>TextRange</i>	No	The text range to which the property is to be applied.
setVal	<i>PropVal</i>	No	The property to apply to the text range.

## SetTextVal

### Description

Sets a specified text property for a text range.

On success, the method does not return anything.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.



Error	Reason
Constants.FE_BadName (-33)	Specified name is illegal.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_BadPropNum (-4)	Specified property number is invalid.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_GenRuleAmbiguous (-55)	General rule in structured document is ambiguous.
Constants.FE_GenRuleConnectorExpected (-57)	General rule in structured document is missing a connector.
Constants.FE_GenRuleItemExpected (-51)	General rule in structured document is missing a rule item.
Constants.FE_GenRuleLeftBracketExpected (-53)	General rule in structured document is missing a left bracket.
Constants.FE_GenRuleMixedConnectors (-52)	General rule in structured document has mixed connectors.
Constants.FE_GenRuleRightBracketExpected (-54)	General rule in structured document is missing a right bracket.
Constants.FE_GenRuleSyntaxError(-56)	General rule in structured document has a syntax error.
Constants.FE_NotTextObject (-41)	Object specified for the text range is not a paragraph (FO_Pgf) or a flow (FO_Flow).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text range could not be found in the specified paragraph or text line.
Constants.FE_OutOfRange (-7)	Specified property value is out of the legal range for the specified property.
Constants.FE_ReadOnly (-6)	Property is read-only and cannot be set.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.

**Returns**

Void

**Syntax**

SetTextVal(textRange, propNum, setVal)

## Parameters

Parameter name	Data Type	Optional	Description
textRange	<i>TextRange</i>	No	The text range to which the property is to be applied.
propNum	int	No	The number of the property to set.
setVal	<i>TypedValue</i>	No	The value of the property to set.

## SilentPrintDoc

### Description

Prints a document or a book using the default print settings. Default print settings are the settings that appear in the Print dialog box when the user attempts to print a document.

The `SilentPrintDoc()` method initializes the print page size and printer name if they do not have values.

To change a document's default print settings, set the document's print properties. [To be linked to the print properties]

For example, to change the turn registration marks on when you print a document, use the `SetInt()` method to set the document's `RegistrationMarks` property.

If you save the document or attempt to print it again within the same session, any changes you make to a document's print settings (except `PrintStartPage` and `PrintEndPage`) appears the next time the user displays the Print dialog box for the document.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_SystemError (-42)	Could not open or close the printer file.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.

### Returns

int

## Syntax

```
SilentPrintDoc()
```

## SimpleImportFormats

### Description

Imports formats from a document to a document or a book. If you import formats to a book, the method imports formats to each book component for which the `ImportFmtInclude` property is set to `True`.

You can OR the following values into the `formatFlags` parameter to specify which formats to import.

Value	Format to Import
Constants.FF_UFF_COLOR (0x0100)	Colors
Constants.FF_UFF_COMBINED_FONTS (0x0800)	Combined Fonts
Constants.FF_UFF_COND (0x0010)	Conditions
Constants.FF_UFF_DOCUMENT_PROPS (0x0400)	Document Properties
Constants.FF_UFF_FONT (0x0002)	Character Catalog Formats
Constants.FF_UFF_MATH (0x0200)	Equation Settings
Constants.FF_UFF_PAGE (0x0004)	Page Layouts
Constants.FF_UFF_PGF (0x0001)	Paragraph Catalog Formats
Constants.FF_UFF_REFPAGE (0x0020)	Reference Pages
Constants.FF_UFF_REMOVE_EXCEPTIONS (0x8000)	Remove exception formats from target documents
Constants.FF_UFF_REMOVE_PAGE_BREAKS (0x4000)	Remove all forced page breaks from target documents
Constants.FF_UFF_TABLE (0x0008)	Table Catalog Formats
Constants.FF_UFF_VAR (0x0040)	Variable Formats
Constants.FF_UFF_XREF (0x0080)	Cross Reference Formats

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker version does not support this operation.
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_Canceled (-44)</code>	User canceled the operation.
<code>Constants.FE_FailedState (-45)</code>	The FrameMaker product failed to open one or more of the book's document files during the import operation.

### Returns

`int`

### Syntax

```
SimpleImportFormats(fromDoc, formatFlags)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>fromDoc</code>	<i>Doc</i>	No	The document or book from which to import formats.
<code>formatFlags</code>	<code>int</code>	No	Bit field specifying which formats to import. Specify 0 for the default flags.

You can OR the following values into the `formatFlags` parameter to specify which formats to import.

Value	Import
<code>Constants.FF_UFF_COLOR (0x100)</code>	Import colors.
<code>Constants.FF_UFF_COMBINED_FONTS (0x800)</code>	Import combined fonts.
<code>Constants.FF_UFF_COND (0x0010)</code>	Import conditions.
<code>Constants.FF_UFF_DOCUMENT_PROPS (0x0400)</code>	Import document properties.

Value	Import
Constants.FF_UFF_FONT (0x0002)	Import Character Catalog formats.
Constants.FF_UFF_MATH (0x200)	Import equation settings.
Constants.FF_UFF_PAGE (0x0004)	Import page layouts.
Constants.FF_UFF_PGF (0x0001)	Import Paragraph Catalog formats.
Constants.FF_UFF_REFPAGE (0x0020)	Import reference pages.
Constants.FF_UFF_REMOVE_EXCEPTIONS (0x8000)	Remove exception formats from target documents.
Constants.FF_UFF_REMOVE_PAGE_BREAKS (0x4000)	Remove all forced page breaks from target documents.
Constants.FF_UFF_TABLE (0x0008)	Import Table Catalog formats.
Constants.FF_UFF_VAR (0x0040)	Import variable formats.
Constants.FF_UFF_XREF (0x0080)	Import cross reference formats.

## SplitElement

### Description

Splits the structural element containing the insertion point into two elements at the insertion point. The insertion point must be inside the element you want to split.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.
Constants.FE_WrongProduct (-60)	Current FrameMaker interface is not Structured FrameMaker.

### Returns

Void

### Syntax

```
SplitElement()
```

## TextLocToElementLoc

### Description

Returns the element location structure that corresponds to the current text location.

On success, the method returns an `ElementLoc` structure containing the element location.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadParameter ()	<code>tloc</code> parameter was empty or was improperly specified
Constants.FE_WrongProduct (-60)	Current FrameMaker interface is not Structured FrameMaker.

### Returns

`ElementLoc`

### Syntax

```
TextLocToElementLoc (tloc)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>tloc</code>	<code>TextLoc</code>	No	The text location structure to convert.

## UnWrapElement

### Description

Removes the selected structural elements, but leaves their contents and child elements intact in the document.

The `UnWrapElement ()` method does not remove all the elements in the selection but only removes the top-level elements.

**NOTE:** At least one structural element must be selected before calling the `UnWrapElement()` method. **NOTE:** The `UnWrapElement()` method has no effect on object elements.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.
Constants.FE_WrongProduct (-60)	Current FrameMaker interface is not Structured FrameMaker.

### Returns

Void

### Syntax

```
UnWrapElement ()
```

### UpdateXRefs

#### Description

Resolves the unresolved the cross-references in a document. The method performs the same operation as clicking Update in the Update References dialog.

You can OR the following values into the `updateXRefFlags` parameter:

Flags	Meaning
Constants.FF_XRUI_FORCE_UPDATE (0x08)	Updates all cross-references, regardless of whether the source document has changed
Constants.FF_XRUI_INTERNAL (0x01)	Only update internal cross-references
Constants.FF_XRUI_OPEN_DOCS (0x02)	Only update cross-references whose sources are in open documents
Constants.FF_XRUI_CLOSED_DOCS (0x04)	Only update cross-references whose sources are in closed documents
Constants.FF_XRUI EVERYTHING (0x01   0x02   0x04)	Update all cross references.

The methodsn returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document
Constants.FE_SomeUnresolved (-22)	There are some unresolved references after the update

**Returns**

int

**Syntax**

UpdateXRefs (updateXRefFlags)

**Parameters**

Parameter name	Data Type	Optional	Description
updateXRefFlags	int	No	Flags that indicate which cross-references to update.

**UpdateDITAResferences****Description**

Updates all DITA references of the specified type.

**Returns**

void

**Syntax**

UpdateDITAResferences (docId, flag);

**Parameters**

Parameter name	Data Type	Optional	Description
docId	int	No	The ID of the document containing the inset.



Parameter name	Data Type	Optional	Description
flag	int	No	The available flags and their values are as follows: FF_DITAUpdateAllConrefs: 0x01 FF_DITAUpdateAllXrefs: 0x02 FF_DITAUpdateAllLinks: 0x04 FF_DITAUpdateAllTopicrefs: 0x08 FF_DITAUpdateAllTopicsetrefs: 0x10 FF_DITAUpdateAllReferences: FF_DITAUpdateAllConrefs   FF_DITAUpdateAllXrefs   FF_DITAUpdateAllLinks   FF_DITAUpdateAllTopicrefs   FF_DITAUpdateAllTopicsetrefs

If `UpdatedDITAResources()` fails, the method assigns following values to `FA_erno`:

Error	Reason
<code>Constants.FE_WrongProduct</code>	Current FrameMaker product doesn't support the operation.
<code>Constants.FE_BadDocId</code>	The Document ID provided is invalid.
<code>Constants.FE_NonDITADocument</code>	The Document provided is not a DITA document.

## SimpleImportElementDefs

### Description

Imports element definitions and the format change list catalog from an EDD or FrameMaker document or book to a FrameMaker document or book.

**NOTE:** If you import element definitions to a book, the `SimpleImportElementDefs()` method imports element definitions to each book component for which the `ImportFmtInclude` property is set to `True`.

You can OR the following flags into the `importFlags` parameter:

Flag	Meaning
<code>Constants.FF_IED_REMOVE_OVERRIDES (0x0001)</code>	Clear format overrides
<code>Constants.FF_IED_REMOVE_BOOK_INFO (0x0002)</code>	Clear formatting inherited from a parent book, if the method is called on a document

Flag	Meaning
Constants.FF_IED_DO_NOT_IMPORT_EDD (0x0004)	Do not treat the target document as an EDD but just import its element catalog, if the target document is an EDD
Constants.FF_IED_NO_NOTIFY (0x0008)	Do not issue the Constants.FA_Note_PreImportElemDefs (64) or Constants.FA_Note_PostImportElemDefs (65) notifications.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document

## Returns

`int`

## Syntax

```
SimpleImportElementDefs (fromDocOrBook, importFlags)
```

## Parameters

Parameter name	Data Type	Optional	Description
<code>fromDocOrBook</code>	<code>Doc   Book</code>	No	The document or book from which the element definitions are to be imported
<code>importFlags</code>	<code>int</code>	No	Specifies how to import formats

## GetUniqueObject

### Description

Gets the object from its persistent unique identifier (UID).

FrameMaker assigns a UID to each object in a document or book that is not identified by a name. The UID, unlike the object's ID, does not change from one session to another.

**NOTE:** When you copy an object to the clipboard and paste it, FrameMaker changes the UID. This also happens when you hide and show conditional text.

The method returns the object, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document
Constants.FE_TypeUnNamed (-19)	Objects of the specifies type are not identified by UIDs
Constants.FE_NameNotFound (-20)	Objects with the specified UID could not be found

## Returns

Object

## Syntax

```
GetUniqueObject(objType, unique)
```

## Parameters

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object (for example, FO_Pgf)
unique	int	No	UID of the object

## Close

### Description

Closes a document or a book.

**NOTE:** If there are unsaved changes in a document and you set `Constants.FF_CLOSE_MODIFIED (1)` for the `flags` parameter, then the `Close()` method abandons the changes and closes the file anyway. **NOTE:** If you set the `flags` parameter to 0, the `Close()` method aborts the `Close` operation and returns `Constants.FE_DocModified (-8)`.

The method returns `FE_Success`, on success.

On failure, the method returns the following value to `FA_errno`:

Error	Reason
Constants.FE_DocModified (-8)	The document was modified and the <code>flags</code> parameter was set to 0

**Returns**

int

**Syntax**

Close(flags)

**Parameters**

Parameter name	Data Type	Optional	Description
flags	int	No	Specifies whether to abort or close open documents or books if they have unsaved changes. Set the Constants.FF_CLOSE_MODIFIED (1) flag to close open documents and books regardless of their state.

**SimpleSave****Description**

Saves a document or book.

If you set the `interactive` parameter to `False` and specify the document or book's current name for `saveAsName`, FrameMaker saves the document or book under its current name.

If you specify another filename for `saveAsName`, FrameMaker saves the document or book to that filename.

If you specify an empty string (`""`), FrameMaker does not save the file. Instead it sets `FA_erro` to `Constants.FE_BadParameter (-43)`. If you set the `interactive` parameter to `True`, FrameMaker displays the Save dialog box and allows the user to choose a filename. The document or book's current name appears as the default name.

The method returns the document it saved, on success.

On failure, the method returns one of the following values to `FA_erro`:

Error	Reason
<ul style="list-style-type: none"> <li>Constants.FE_Canceled (-44) Constants.FE_Canceled-ByClient (-86)</li> </ul>	<ul style="list-style-type: none"> <li>Constants.FV_FileNotWritable (33): File was not writable.</li> <li>Constants.FV_BadSaveFileName (34): The specified file name is not allowed by the operating system.</li> <li>Constants.FV_BadFileId (35): The file's operating system ID was bad.</li> <li>Constants.FV_CancelSaveFileIsInUse (50): The file is in use and the user did not or could not reset the lock. Or the file is in use, and the Constants.FS_FileIsInUse (17) parameter is set to Constants.FV_DoCancel (0), or it is set to Constants.FV_ResetLockAndContinue (7) but FrameMaker could not reset the lock.</li> <li>Constants.FV_CancelSaveModDateChanged (52): The file has changed since the last time it was opened or saved in the current session.</li> <li>Constants.FV_LockCouldntBeReset (2): File lock could not be reset.</li> <li>Constants.FV_LockWasReset (0): File lock was reset.</li> <li>Constants.FV_LockNotReset (1): File lock was not reset.</li> <li>Constants.FV_FileIsViewOnly (4): File was View Only.</li> <li>Constants.FV_NonPortableSaveName (37): File had a name that is not portable (Macintosh only).</li> <li>Constants.FV_NonPortableFileRefs (38): Referenced graphics files had filenames that are not portable (Macintosh only).</li> </ul>
Constants.FE_WrongProduct (-60)	The specified file contains structure and the current product interface is not Structured FrameMaker.
Constants.FE_FailedState (-45) Constants.FE_BadParameter (-43)	The filename was invalid.
Constants.FE_FilterFailed (-89)	Constants.FV_InvalidSaveFilter (42): The filter specified by Constants.FS_SaveFileTypeHint (16) is not installed, or the syntax for Constants.FS_SaveFileTypeHint (16) is invalid.

## Returns

*Doc*

## Syntax

SimpleSave (saveAsName, interactive)

## Parameters

Parameter name	Data Type	Optional	Description
saveAsName	string	No	The absolute path in which to save the document or book.
interactive	int	No	Specifies whether the FrameMaker product displays messages and warnings to the user. Setting this to True displays messages and warnings.

## NewGraphicObject

### Description

Creates the following types of graphic objects:

- Arc
- Ellipse
- Flow
- Group
- Inset
- Line
- Math
- Polyline
- Polygon
- Rectangle
- RoundRect
- TextFrame
- TextLine
- UnanchoredFrame

**NOTE:** To create an anchored frame, use `NewAnchoredObject()`.

If there is more than one object within the parent frame, the `NewGraphicObject()` method adds the new ESTK graphic object to the end of the linked list of child objects. That is, it puts it in the front of the back-to-front draw order. The method automatically takes care of updating the object's `PrevGraphicInFrame` and `NextGraphicInFrame` properties. The `NewGraphicObject()` method gives the new method graphic object a set of arbitrary default properties.

The method returns the newly created graphic object, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_NotFrame (-12)	Specified parent object is not a frame.
Constants.FE_BadNew (-23)	Object cannot be created.

**Returns**

Object

**Syntax**

`NewGraphicObject(objType, parent)`

**Parameters**

Parameter name	Data Type	Optional	Description
objType	int	No	The type of ESTK graphic object to create (for example, FO_Rectangle or FO_Line).
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the object.

**NewArc****Description**

Creates an arc object.

See [NewGraphicObject](#) for more information.

**Returns**

[Arc](#)

**Syntax**

`NewArc(parent)`

## Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the arc.

## NewEllipse

### Description

Creates an ellipse object.

See [NewGraphicObject](#) for more information.

### Returns

[Ellipse](#)

### Syntax

NewEllipse (parent)

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the ellipse.

## NewFlow

### Description

Creates a flow object.

See [NewGraphicObject](#) for more information.



**Returns***Flow***Syntax**

NewFlow (parent)

**Parameters**

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the flow.

**NewGroup****Description**

Creates a group object.

See [NewGraphicObject](#) for more information.**Returns***Group***Syntax**

NewGroup (parent)

**Parameters**

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the group.

## NewInlineComponentOfType

### Description

Creates an inline component. Presently only one type of inline component exists, and that is the mini-TOC.

### Returns

Void

### Syntax

```
NewInlineComponentOfType (inlineCompType, tags, hyperLinks, textLocP)
```

### Parameters

Parameter name	Data Type	Optional	Description
inlineCompType	int	No	Type of inline component. Presently only one type exists: <code>Constants.FV_MiniTOC</code>
tags	<i>Strings</i>	No	Paragraph tags to include for the inline component
hyperLinks	Constants.FP_InsertLinks	No	Flag that specifies if hyperlinks need to be created in the inline component.
textLocP	<i>TextLoc</i>	No	Text location in docId at which to create the inline component.

## NewInset

### Description

Creates an inset object.

See [NewGraphicObject](#) for more information.

### Returns

*Inset*

### Syntax

```
NewInset (parent)
```

## Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the inset.

## NewLine

### Description

Creates a line object.

See [NewGraphicObject](#) for more information.

### Returns

[Line](#)

### Syntax

NewLine (parent)

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the line.

## NewMath

### Description

Creates a math object.

See [NewGraphicObject](#) for more information.

**Returns***Math***Syntax**

NewMath (parent)

**Parameters**

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the math object.

**NewMathML****Description**

Creates a MathML object.

See *NewGraphicObject* for more information.**Returns***MathML***Syntax**

NewMathML (parent)

**Parameters**

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the MathML object.

## NewPolyline

### Description

Creates a polyline object.

See [NewGraphicObject](#) for more information.

### Returns

*Polyline*

### Syntax

NewPolyline (parent)

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the polyline.

## NewPolygon

### Description

Creates a polygon object.

See [NewGraphicObject](#) for more information.

### Returns

*Polygon*

### Syntax

NewPolygon (parent)

## Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the polygon.

## NewRectangle

### Description

Creates a rectangle object.

See [NewGraphicObject](#) for more information.

### Returns

*Rectangle*

### Syntax

NewRectangle (parent)

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the rectangle.

## NewRoundRect

### Description

Creates a round rectangle object.

See [NewGraphicObject](#) for more information.

**Returns**[RoundRect](#)**Syntax**

NewRoundRect (parent)

**Parameters**

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the round rectangle.

**NewTextFrame****Description**

Creates a text frame object.

See [NewGraphicObject](#) for more information.**Returns**[TextFrame](#)**Syntax**

NewTextFrame (parent)

**Parameters**

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the text frame.

## NewTextLine

### Description

Creates a text line object.

See [NewGraphicObject](#) for more information.

### Returns

*TextLine*

### Syntax

NewTextLine (parent)

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the text line.

## NewUnanchoredFrame

### Description

Creates an unanchored frame object.

See [NewGraphicObject](#) for more information.

### Returns

*UnanchoredFrame*

### Syntax

NewUnanchoredFrame (parent)



## Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMa ker Graphic Object	No	The parent frame in which to create the unanchored frame.

## NewAnchoredFormattedObject

### Description

Creates the following types of anchored objects:

- Var
- XRef
- Tbl

`NewAnchoredFormattedObject()` inserts the object at the specified location in text. It uses arbitrary default properties for the new object.

If you call `NewAnchoredFormattedObject()` to create a table, it uses the default numbers of rows and columns from the specified Table Catalog format. To use the default Table Catalog format for a new table, set the format to `Null`. To specify the number of rows and columns when you create a table, use `NewTable()`.

`NewAnchoredFormattedObject()` returns the created anchor object, on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId(-2)</code> )	Invalid document.
<code>Constants.FE_BadObjId(-3)</code> )	Invalid object.
<code>Constants.FE_NotTextObject(-41)</code>	Object specified for text location is not a paragraph ( <code>FO_Pgf</code> ).
<code>Constants.FE_OffsetNotFound(-21)</code>	Offset specified for the text location could not be found in the specified paragraph or text line.
<code>Constants.FE_BadNew(-23)</code> )	Object cannot be created; the format specified by <code>format</code> may not exist.

**Returns**

Object

**Syntax**

```
NewAnchoredFormattedObject(objType, format, textLoc)
```

**Parameters**

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object to create (for example, FO_XRef).
format	string	No	The string that specifies the object's format (for example, Heading & Page for a cross-reference, Format A for a table, or Current Date (Long) for a variable).
textLoc	<i>TextLoc</i>	No	The text location at which to insert the anchored object.

**NewAnchoredFormattedVar****Description**

Creates an `AnchoredFormattedVar` object.

See [NewAnchoredFormattedObject](#) under the Doc class for more information.

**Returns***Var***Syntax**

```
NewAnchoredFormattedVar(format, textLoc)
```

## Parameters

Parameter name	Data Type	Optional	Description
format	string	No	The string that specifies the object's format (for example, <code>Heading &amp; Page</code> for a cross-reference, <code>Format A</code> for a table, or <code>Current Date (Long)</code> for a variable).
textLoc	<i>TextLoc</i>	No	The text location at which to insert the <code>AnchoredFormattedVar</code> object.

## NewAnchoredFormattedXRef

### Description

Creates an `AnchoredFormattedXRef` object.

See [NewAnchoredFormattedObject](#) under the Doc class for more information.

### Returns

*XRef*

### Syntax

```
NewAnchoredFormattedXRef(format, textLoc)
```

### Parameters

Parameter name	Data Type	Optional	Description
format	string	No	The string that specifies the object's format (for example, <code>Heading &amp; Page</code> for a cross-reference, <code>Format A</code> for a table, or <code>Current Date (Long)</code> for a variable).
textLoc	<i>TextLoc</i>	No	The text location at which to insert the <code>AnchoredFormattedXRef</code> object.

## NewAnchoredFormattedTbl

### Description

Creates an `AnchoredFormattedTbl` object.

See [NewAnchoredFormattedObject](#) under the Doc class for more information.

## Returns

*Tbl*

## Syntax

```
NewAnchoredFormattedTbl (format, textLoc)
```

## Parameters

Parameter name	Data Type	Optional	Description
format	string	No	The string that specifies the object's format (for example, <code>Heading &amp; Page</code> for a cross-reference, <code>Format A</code> for a table, or <code>Current Date (Long)</code> for a variable).
textLoc	<i>TextLoc</i>	No	The text location at which to insert the <code>AnchoredFormattedTbl</code> object.

## NewAnchoredObject

### Description

Creates any of the following anchored objects:

- AFrame
- Fn
- Marker
- TiApiClient
- Table

`NewAnchoredObject ()` inserts the object at the specified location in text. It uses arbitrary default properties for the new object. Tables created by `NewAnchoredObject ()` have a single column and a single body row. It is usually easier to use `NewTable ()` to create tables.

The method returns the created anchored object on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadNew(-23)	Object cannot be created.

Error	Reason
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadOperation (-27)	Method call specified a bad operation.
Constants.FE_NotTextObject (-41)	Object specified for text location is not a paragraph (FO_Pgf).
Constants.FE_OffsetNotFound(-21)	Offset specified for the text location could not be found in the specified paragraph or text line.

**Returns**

Object

**Syntax**

`NewAnchoredObject(objType, textloc)`

**Parameters**

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object to create (for example, FO_Marker or FO_Fn).
textloc	<i>TextLoc</i>	No	The text location at which to insert the anchored object.

**NewAnchoredAFrame****Description**

Creates an anchored AFrame object.

See [NewAnchoredObject](#) under the Doc class for more information.

**Returns**

*AFrame*

**Syntax**

`NewAnchoredAFrame(textloc)`

## Parameters

Parameter name	Data Type	Optional	Description
textloc	<i>TextLoc</i>	No	The text location at which to insert the anchored AFrame.

## NewAnchoredFn

### Description

Creates an anchored Fn object.

See [NewAnchoredObject](#) under the Doc class for more information.

### Returns

*Fn*

### Syntax

```
NewAnchoredFn(textloc)
```

### Parameters

Parameter name	Data Type	Optional	Description
textloc	<i>TextLoc</i>	No	The text location at which to insert the anchored Fn object.

## NewAnchoredMarker

### Description

Creates an anchored Marker object.

See [NewAnchoredObject](#) under the Doc class for more information.

### Returns

*Marker*

### Syntax

```
NewAnchoredMarker(textloc)
```

## Parameters

Parameter name	Data Type	Optional	Description
textloc	<i>TextLoc</i>	No	The text location at which to insert the anchored Marker.

## NewAnchoredTiApiClient

### Description

Creates an anchored TiApiClient object.

See [NewAnchoredObject](#) under the Doc class for more information.

### Returns

*TiApiClient*

### Syntax

```
NewAnchoredTiApiClient(textloc)
```

### Parameters

Parameter name	Data Type	Optional	Description
textloc	<i>TextLoc</i>	No	The text location at which to insert the anchored TiApiClient object.

## NewAnchoredTbl

### Description

Creates an anchored table object.

See [NewAnchoredObject](#) under the Doc class for more information.

### Returns

*Tbl*

### Syntax

```
NewAnchoredTbl(textloc)
```

## Parameters

Parameter name	Data Type	Optional	Description
textloc	<i>TextLoc</i>	No	The text location at which to insert the anchored Table.

## NewSeriesObject

### Description

Creates the following series objects:

- BodyPage
- BookComponent
- Pgf

The method allows you to specify the position in the series at which to add the new object.

The method returns the created series object, on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadNew (-23)	Object cannot be created.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_NotBodyPage (-24)	The <code>prevId</code> parameter must specify a body page.
Constants.FE_NotBodyPgf (-25)	The <code>prevId</code> parameter must specify a paragraph..
Constants.FE_NotBookComponent (-26)	The <code>prevId</code> parameter must specify a book component..

### Returns

One of the following objects: Body page, Pgf and Book Component

### Syntax

```
NewSeriesObject(objType, previous)
```



## Parameters

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object to create (for example, FO_BodyPage or FO_Pgf).
previous	One of: Body page, Pgf and Book Component	No	The object in the series after which to add the new object. To add a paragraph at the start of a flow, specify the flow. To add an object at the beginning of any other series, specify 0.

## NewSeriesBodyPage

### Description

Creates a body page.

See [NewSeriesObject](#) under the Doc class for more information.

### Returns

[BodyPage](#)

### Syntax

NewSeriesBodyPage(previous)

### Parameters

Parameter name	Data Type	Optional	Description
previous	BodyPage	No	The object in the series after which to add the new body page. To add a paragraph at the start of a flow, specify the flow. To add a body page at the beginning of any other series, specify 0.

## NewSeriesPgf

### Description

Creates a paragraph.

See [NewSeriesObject](#) under the Doc class for more information.

### Returns

*Pgf*

### Syntax

NewSeriesPgf(previous)

### Parameters

Parameter name	Data Type	Optional	Description
previous	Pgf	No	The object in the series after which to add the new paragraph. To add a paragraph at the start of a flow, specify the flow. To add a paragraph at the beginning of any other series, specify 0.

## GetNamedObject

### Description

Gets the object with the specified name and type. The method works with the following objects:

- AttrCondExpr
- CharFmt
- Color
- CombinedFontDefn
- Command
- CondFmt
- ElementDef
- FmtChangeList
- Menu
- MenuItemSeparator
- MasterPage
- PgfFmt

- RefPage
- RulingFmt
- TblFmt
- UnanchoredFrame (reference frame)
- VarFmt
- XRefFmt
- MarkerType

The method returns the object, on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_TypeUnNamed (-19)	Objects of the specified type do not have names.
Constants.NameNotFound (-20)	Object with the specified name and type does not exist in the specified document.

### Returns

Object

### Syntax

`GetNamedObject(objType, name)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>objType</code>	<code>int</code>	No	The type of object (for example <code>FO_TblFmt</code> ).
<code>name</code>	<code>string</code>	No	The name of the object you want to get.

## GetNamedAttrCondExpr

### Description

Gets an attribute conditional expression object.

See [GetNamedObject](#) under the Doc class for more information.

**Returns**[AttrCondExpr](#)**Syntax**

GetNamedAttrCondExpr (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the AttrCondExpr object you want to get.

**GetNamedCharFmt****Description**

Gets a Character Format object.

See [GetNamedObject](#) under the Doc class for more information.**Returns**[CharFmt](#)**Syntax**

GetNamedCharFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the CharFmt object you want to get.

**GetNamedColor****Description**

Gets a Color object.

See [GetNamedObject](#) under the Doc class for more information.

**Returns***Color***Syntax**

GetNamedColor (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the Color object you want to get.

**GetNamedCombinedFontDefn****Description**

Gets a Combined Font Definition object.

See [GetNamedObject](#) under the Doc class for more information.**Returns***CombinedFontDefn***Syntax**

GetNamedCombinedFontDefn (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the CombinedFontDefn object you want to get.

**GetNamedCondFmt****Description**

Gets a Conditional Format object.

See [GetNamedObject](#) under the Doc class for more information.

**Returns***CondFmt***Syntax**

GetNamedCondFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the CondFmt object you want to get.

**GetNamedElementDef****Description**

Gets an Element Definition object.

See [GetNamedObject](#) under the Doc class for more information.**Returns***ElementDef***Syntax**

GetNamedElementDef (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the ElementDef object you want to get.

**GetNamedFmtChangeList****Description**

Gets a Format Change List object.

See [GetNamedObject](#) under the Doc class for more information.

**Returns***FmtChangeList***Syntax**

GetNamedFmtChangeList (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the FmtChangeList object you want to get.

**GetNamedMasterPage****Description**

Gets a MasterPage object.

See [GetNamedObject](#) under the Doc class for more information.**Returns***MasterPage***Syntax**

GetNamedMasterPage (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the MasterPage object you want to get.

**GetNamedPgFmt****Description**

Gets a Paragraph Format object.

See [GetNamedObject](#) under the Doc class for more information.

**Returns**[PgFmt](#)**Syntax**

GetNamedPgFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the PgFmt object you want to get.

**GetNamedRefPage****Description**

Gets a Reference Page object.

See [GetNamedObject](#) under the Doc class for more information.**Returns**[RefPage](#)**Syntax**

GetNamedRefPage (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the RefPage object you want to get.

**GetNamedRulingFmt****Description**

Gets a Ruling Format object.

See [GetNamedObject](#) under the Doc class for more information.



**Returns***RulingFmt***Syntax**

GetNamedRulingFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the RulingFmt object you want to get.

**GetNamedTblFmt****Description**

Gets a Table Format object.

See [GetNamedObject](#) under the Doc class for more information.**Returns***TblFmt***Syntax**

GetNamedTblFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the TblFmt object you want to get.

**GetNamedUnanchoredFrame****Description**

Gets an Unanchored Frame object.

See [GetNamedObject](#) under the Doc class for more information.

**Returns**[UnanchoredFrame](#)**Syntax**

GetNamedUnanchoredFrame (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the Unanchored Frame object you want to get.

**GetNamedVarFmt****Description**

Gets a Variable Format object.

See [GetNamedObject](#) under the Doc class for more information.**Returns**[VarFmt](#)**Syntax**

GetNamedVarFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the Variable Format object you want to get.

**GetNamedXRefFmt****Description**

Gets a Cross Reference Format object.

See [GetNamedObject](#) under the Doc class for more information.

**Returns***XRefFmt***Syntax**

GetNamedXRefFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the Cross Reference Format object you want to get.

**GetNamedMarkerType****Description**

Gets a Marker Type object.

See [GetNamedObject](#) under the Doc class for more information.**Returns***MarkerType***Syntax**

GetNamedMarkerType (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the Marker Type object you want to get.

**NewNamedObject****Description**

Creates the following named objects:

- AttrCondExpr

- CharFmt
- CombinedFontDefn
- Color
- Command
- CondFmt
- ElementDef
- FmtChangeList
- MasterPage
- Menu
- MenuItemSeparator
- PgfFmt
- RefPage
- RulingFmt
- TblFmt
- VarFmt
- XRefFmt
- MarkerType

This method uses arbitrary default properties for the objects it creates.

**NOTE:** When you create a new element definition, it does not appear in the Element Catalog unless you set `ElementInCatalog` to `True`. When you create a new book and specify a pathname, you must specify an absolute pathname for the name argument. To create an untitled book, pass an empty string for the name argument.

`NewNamedObject()` returns the created named object on success.

If this method fails, the method assigns one of the following values to `FA_errno`.

FA_errno value	Meaning
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_BadName</code> (-3 3)	Specified name for the new object is invalid.
<code>Constants.FE_BadNew</code> (-23)	Object cannot be created.
<code>Constants.FE_DupName</code> (-3 2)	Specified name for the new object belongs to an existing object.

### Returns

Object of type `objType`

## Syntax

```
NewNamedObject(objType, name)
```

## Parameters

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object to create (for example, Constants.FO_MasterPage or Constants.FO_PgfFmt).
name	string	No	The name to be assigned to the created object.

## NewNamedAttrCondExpr

### Description

Creates a named Attribute Conditional Expression.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[AttrCondExpr](#)

## Syntax

```
NewNamedAttrCondExpr(name)
```

## Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Attribute Conditional Expression.

## NewNamedCharFmt

### Description

Creates a named Character Format.

See [NewNamedObject](#) under the Doc class for more information.

**Returns**[CharFmt](#)**Syntax**

NewNamedCharFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Character Format.

**NewNamedCombinedFontDefn****Description**

Creates a named Combined Font Definition.

See [NewNamedObject](#) under the Doc class for more information.**Returns**[CombinedFontDefn](#)**Syntax**

NewNamedCombinedFontDefn (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Combined Font Definition.

**NewNamedColor****Description**

Creates a named Color.

See [NewNamedObject](#) under the Doc class for more information.

**Returns***Color***Syntax**

NewNamedColor (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Color.

**NewNamedCondFmt****Description**

Creates a named Conditional Format.

See [NewNamedObject](#) under the Doc class for more information.**Returns***CondFmt***Syntax**

NewNamedCondFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Conditional Format.

**NewNamedElementDef****Description**

Creates a named Element Definition.

See [NewNamedObject](#) under the Doc class for more information.

**Returns***ElementDef***Syntax**`NewNamedElementDef (name)`**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Element Definition.

**NewNamedFmtChangeList****Description**

Creates a named Format Change List.

See [NewNamedObject](#) under the Doc class for more information.

**Returns***FmtChangeList***Syntax**`NewNamedFmtChangeList (name)`**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Format Change List.

**NewNamedMasterPage****Description**

Creates a named Master Page.

See [NewNamedObject](#) under the Doc class for more information.



**Returns***MasterPage***Syntax**

NewNamedMasterPage (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Master Page.

**NewNamedPgFmt****Description**

Creates a named Paragraph Format.

See [NewNamedObject](#) under the Doc class for more information.**Returns***PgfFmt***Syntax**

NewNamedPgFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Paragraph Format.

**NewNamedRefPage****Description**

Creates a named Reference Page.

See [NewNamedObject](#) under the Doc class for more information.

**Returns**[RefPage](#)**Syntax**

NewNamedRefPage (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Reference Page.

**NewNamedRulingFmt****Description**

Creates a named Ruling Format.

See [NewNamedObject](#) under the Doc class for more information.**Returns**[RulingFmt](#)**Syntax**

NewNamedRulingFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Ruling Format.

**NewNamedTblFmt****Description**

Creates a named Table Format.

See [NewNamedObject](#) under the Doc class for more information.

**Returns***TblFmt***Syntax**

NewNamedTblFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Table Format.

**NewNamedVarFmt****Description**

Creates a named Variable Format.

See [NewNamedObject](#) under the Doc class for more information.**Returns***VarFmt***Syntax**

NewNamedVarFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Variable Format.

**NewNamedXRefFmt****Description**

Creates a named Cross Reference Format.

See [NewNamedObject](#) under the Doc class for more information.

**Returns***XRefFmt***Syntax**

NewNamedXRefFmt (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Cross Reference Format.

**NewNamedMarkerType****Description**

Creates a named Marker Type.

See [NewNamedObject](#) under the Doc class for more information.**Returns***MarkerType***Syntax**

NewNamedMarkerType (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Marker Type.

**DeleteUnusedFmts****Description**

Deletes unused formats (character, paragraph or table) from the document.

The method returns `Constants.FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadDelete (-17)</code>	The format to be deleted is not one of character, paragraph or table.

### Returns

`int`

### Syntax

`DeleteUnusedFmts (objType)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>objType</code>	<code>int</code>	No	The type of format to be deleted.

## DeleteUnusedCharFmt

### Description

Deletes the unused Character Formats.

See [DeleteUnusedFmts](#) under the Doc class for more information.

### Returns

`int`

### Syntax

`DeleteUnusedCharFmt ()`

## DeleteUnusedPgFmt

### Description

Deletes the unused Paragraph Formats.

See [DeleteUnusedFmts](#) under the Doc class for more information.

**Returns**

int

**Syntax**

DeleteUnusedPgfmt ()

**DeleteUnusedTblFmt****Description**

Deletes the unused Table Formats.

See [DeleteUnusedFmts](#) under the Doc class for more information.**Returns**

int

**Syntax**

DeleteUnusedTblFmt ()

**Find****Description**

Performs the same actions as using the Find dialog box to search a document for text or other types of content.

The properties you can assign to `findParamsp` are as follows:

Property	Meaning and Possible Values
Constants.FS_FindText (1)	The text to search for.
Constants.FS_FindElementTag (2)	Strings as follows: <pre>propVal.u.ssva1.len = FV_NumFindElementItems; propVal.u.ssva1.val[FV_FindElemTag] = [an_element_tag]; propVal.u.ssva1.val[FV_FindAttrName]= [an_attribute_name]; propVal.u.ssva1.val[FV_FindAttrValue] = [an_attribute_value];</pre> All of the strings must be present, but any or all may be empty.

Property	Meaning and Possible Values
Constants.FS_FindCharFmt (3)	<p>No associated property. One or more of the following additional properties should be specified to tailor the search.</p> <ul style="list-style-type: none"> <li>• FontFamily</li> <li><i>CombinedFont</i></li> <li><i>FontSize</i></li> <li><i>FontAngle</i></li> <li><i>FontWeight</i></li> <li><i>FontVariation</i></li> <li><i>Color</i></li> <li><i>Spread</i></li> <li><i>Stretch</i></li> <li><i>Language</i></li> <li><i>Underline</i></li> <li><i>Overline</i></li> <li><i>Strikethrough</i></li> <li><i>ChangeBar</i></li> <li><i>Capitalization</i></li> <li><i>Position</i></li> <li><i>Tsume</i></li> </ul>
Constants.FS_FindPgfTag (4)	Paragraph tag.
Constants.FS_FindCharTag (5)	Character tag.
Constants.FS_FindTableTag (6)	Table tag.
Constants.FS_FindObject (8)	<p>Value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_FindAnyMarker (0)</li> <li><i>Constants.FV_FindAnyXRef (1)</i></li> <li><i>Constants.FV_FindUnresolvedXRef (2)</i></li> <li><i>Constants.FV_FindAnyTextInset (3)</i></li> <li><i>Constants.FV_FindUnresolvedTextInset (4)</i></li> <li><i>Constants.FV_FindAnyPub (5)</i></li> <li><i>Constants.FV_FindAnyVariable (6)</i></li> <li><i>Constants.FV_FindAnchoredFrame (7)</i></li> <li><i>Constants.FV_FindFootnote (8)</i></li> <li><i>Constants.FV_FindAnyTable (9)</i></li> <li><i>Constants.FV_FindAutomaticHyphen (10)</i></li> <li><i>Constants.FV_FindAnyRubi (11)</i></li> </ul>

Property	Meaning and Possible Values
Constants.FS_FindMarkerOfType (9)	Marker type.
Constants.FS_FindMarkerText (10)	Marker text.
Constants.FS_FindXRefWithFormat (11)	Format string.
Constants.FS_FindNamedVariable (12)	Variable name.
Constants.FS_FindCondTextInCondTags (13)	Condition tags.
Constants.FS_FindCondTextNotInCondTags (14)	Condition tag.
Constant.FS_FindCustomizationFlags (15)	<p>An optional parameter of type <code>FT_Integer</code> that may be any of the following bit flags OR'ed together:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FF_FIND_CONSIDER_CASE (0x01)</code></li> <li>• <code>Constants.FF_FIND_WHOLE_WORD (0x02)</code></li> <li>• <code>Constants.FF_FIND_USE_WILDCARDS (0x04)</code></li> <li>• <code>Constants.FF_FIND_BACKWARDS (0x08)</code></li> <li>• <code>Constants.FF_FIND_USE_REGEX (0x16)</code></li> </ul> <p>If no customization flags are specified, the default is to search forward, to not use wildcards, to not consider case, and to not use whole words.</p>
Constants.FS_FindWrap (16)	<p>A flag that determines whether the find operation wraps when it reaches the location where the search began. Default is <code>True</code>; the find operation wraps. If <code>False</code>, after reaching the location where the search began, the find operation returns an empty <code>TextRange</code> and <code>FA_errno</code> is set to <code>Constants.FE_NotFound (-95)</code>.</p>

Whenever the `Find()` method finds something that corresponds to a text range (a word, object anchor, marker, etc.), it returns a `TextRange` structure for that range. However, when searching for structure elements, you can find elements that have no corresponding text range. Structure elements for the following table parts have no corresponding text range:

- Table title
- Table head
- Table foot
- Table body



- Table row
- Table cell

When the `Find()` method finds a structure element for one of these objects, it returns an empty `TextRange` structure and sets `FA_errno` to `FE_Success`. In this case, you can get the document's `ElementSelection` property to return a corresponding `ElementRange` structure for the table part structure element.

On failure, an empty text range is returned and `FA_errno` is set to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadParameter (-43)	<code>findParams</code> was empty or a parameter was improperly specified.
Constants.FE_BadInsertPos (-65)	The value of the <code>textLoc</code> parameter is invalid.
Constants.FE_NotTextObject (-41)	<code>textLoc</code> parameter is not a text location.

## Returns

*TextRange*

## Syntax

```
Find(textLoc, findParams)
```

## Parameters

Parameter name	Data Type	Optional	Description
<code>textLoc</code>	<i>TextLoc</i>	No	The text location to begin searching from.
<code>findParams</code>	<i>PropVals</i>	No	A property list that specifies what to search for.

## SetElementRange

### Description

Sets an element range property.

To specify a selection that includes the root element, set `beg.parentId` to 0, `beg.childId` to the root element, and `end.childId` to 0.

The method does not return anything on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_BadObjId</code> (-3)	invalid object.
<code>Constants.FE_WrongProduct</code> (-60)	Current product interface is not Structured FrameMaker.

### Returns

Void

### Syntax

```
SetElementRange(propNum, setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>propNum</code>	<code>int</code>	No	The property to set. Specify an ESTK defined property, such as <code>ElementSelection</code> .
<code>setVal</code>	<i>Element Range</i>	No	The element range to which the property must be set.

## ElementLocToTextLoc

### Description

Returns the text location structure that corresponds with the current element location.

The method returns the text location structure containing the element location, on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.

Error	Reason
<code>Constants.FE_Badparameter (-43)</code>	The <code>eloc</code> parameter was empty or incorrectly specified.
<code>Constants.FE_WrongProduct (-60)</code>	Current product interface is not Structured FrameMaker.

**Returns***TextLoc***Syntax**`ElementLocToTextLoc (eloc)`**Parameters**

Parameter name	Data Type	Optional	Description
<code>eloc</code>	<i>ElementLoc</i>	No	The element location structure to convert.

**UpdateVariables****Description**

Updates all the variables in a document. The method performs the same operation as clicking Update in the Variable dialog box.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_WrongProduct (-60)</code>	Product does not support the operation.
<code>Constants.FE_SystemError (-42)</code>	Could not allocate memory.

**Returns**

int

**Syntax**

UpdateVariables()

**TrackChangesAcceptAll****Description**

Accepts all the track changes in the specified document.

The method returns `FE_Success` on success and returns a non-zero integer on failure.

Returns `Constants.FE_AcceptRejectCalledOnWrongPreviewState` (-110) if the document is not in `Preview Off` state.

**Returns**

int

**Syntax**

TrackChangesAcceptAll()

**TrackChangesRejectAll****Description**

Rejects all the track changes in the specified document.

The method returns `FE_Success` on success and returns a non-zero integer on failure.

Returns `Constants.FE_AcceptRejectCalledOnWrongPreviewState` (-110) if the document is not in `Preview Off` state.

**Returns**

int

**Syntax**

TrackChangesRejectAll()

## UpdateXRef

### Description

Updates the cross-references in a document. It performs the same operation as clicking Update in the Cross-Reference window.

You can OR the values listed in the following tables into the `updateXRefFlags` argument.

Value	What to update
<code>Constants.FF_XRUI_INTERNAL (0x01)</code>	Updates only internal cross-references.
<code>Constants.FF_XRUI_OPEN_DOCUMENTS (0x02)</code>	Updates only cross-references whose sources are in open documents.
<code>Constants.FF_XRUI_CLOSED_DOCUMENTS (0x04)</code>	Updates only cross-references whose sources are in closed documents.
<code>Constants.FF_XRUI_FORCE_UPDATE (0x08)</code>	Updates all cross-references, regardless of whether the source document has changed.
<code>Constants.FF_XRUI EVERYTHING (0x01 0x02 0x04)</code>	Updates all cross references.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_WrongProduct (-60)</code>	Product does not support the operation.
<code>Constants.FE_SomeUnresolved (-42)</code>	There are some unresolved references after the update.

### Returns

`int`

### Syntax

```
UpdateXRef(srcDoc, xref)
```

## Parameters

Parameter name	Data Type	Optional	Description
srcDoc	<i>Doc</i>	No	The document in which to update cross-references.
xref	<i>XRef</i>	No	Flags to indicate which cross-references to update. .

## UpdateDITAReference

### Description

Updates a DITA object

Error	Reason
<code>Constants.FE_WrongProduct</code>	Current FrameMaker product doesn't support the operation.
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadElementId</code>	The Element ID provided is invalid.
<code>Constants.FE_NonDITADocument</code>	The document provided is not a DITA document.
<code>Constants.FE_BadParameter</code>	The <code>objType</code> provided is invalid or the <code>objType</code> is not valid for the type of DITA document provided.
<code>Constants.FE_UpdatedDITAReferenceFailedInvalidElementType</code>	Update operation failed because either the element specified is not a reference type of element or it does not match the specified object type.
<code>Constants.FE_UpdatedDITAReferenceFailedCannotResolveReference</code>	Update operation failed because the reference cannot be resolved.
<code>Constants.FE_UpdatedDITAReferenceFailedCannotFindReferencedFile</code>	Update operation failed because the referenced file cannot be found at the specified location.

Error	Reason
<code>Constants.FE_UpdateDITAR eferenceFailedCannotOpen ReferencedFile</code>	Update operation failed because the referenced file cannot be opened from the specified location.
<code>Constants.FE_UpdateDITAR eferenceFailedCannotConv ertToFMObject</code>	Update operation failed because the corresponding FM object cannot be created.
<code>Constants.FE_UpdateDITAR eferenceFailed</code>	Update operation failed.

## UpdateDITARefereces

### Description

Updates all DITA references of the specified type.

### Returns

void

### Syntax

```
UpdateDITARefereces(docId, flag);
```

### Parameters

Parameter name	Data Type	Optional	Description
docId	int	No	The ID of the document containing the inset.
flag	int	No	The available flags and their values are as follows: FF_DITAUpdateAllConrefs: 0x01 FF_DITAUpdateAllXrefs: 0x02 FF_DITAUpdateAllLinks: 0x04 FF_DITAUpdateAllTopicrefs: 0x08 FF_DITAUpdateAllTopicsetrefs: 0x10 FF_DITAUpdateAllReferences: FF_DITAUpdateAllConrefs   FF_DITAUpdateAllXrefs   FF_DITAUpdateAllLinks   FF_DITAUpdateAllTopicrefs   FF_DITAUpdateAllTopicsetrefs

If `UpdateDITAResources()` fails, the method assigns following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct</code>	Current FrameMaker product doesn't support the operation.
<code>Constants.FE_BadDocId</code>	The Document ID provided is invalid.
<code>Constants.FE_NonDITADocument</code>	The Document provided is not a DITA document.

## Save

### Description

Saves a document.

See [Save](#) under the `Book` class for more information.

### Returns

Object

### Syntax

```
Save(saveAsName, saveParams, saveReturnParams)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>saveAsName</code>	<code>string</code>	No	The path for saving the document or book.
<code>saveParams</code>	<code>PropVals</code>	No	A property list that tells FrameMaker how to save the file and how to respond to errors and other conditions. Use <code>GetSaveDefaultParams()</code> or <code>AllocatePropVals()</code> to create and allocate memory for this property list. To use the default list, specify <code>null</code> .
<code>saveReturnParams</code>	<code>PropVals</code>	No	A property list that returns information about how the FrameMaker product saved the file.



## Import

### Description

Imports text or graphics into a document.

See [Import](#) under the Book class for more information.

### Returns

Object

### Syntax

```
Import(textLocP, filename, importParams, importReturnParams)
```

### Parameters

Parameter name	Data Type	Optional	Description
textLocP	<i>TextLoc</i>	No	The text location at which to import the file.
filename	string	No	The full path of the file to import.
importParams	<i>PropVals</i>	No	A property list specifying how to import the file and how to respond to errors and other conditions. To use the default list, specify null.
importReturnParams	<i>PropVals</i>	No	A property list that provides information about how FrameMaker imported the file. This parameter must be initialized before you call Import().

## AddNewBuildExpr

### Description

Adds a Boolean conditional expression to the document.

The method returns `FE_Success`, on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.

Error	Reason
<code>Constants.FE_ReadOnly</code> (-6)	Document is read-only.
<code>Constants.FE_BadName</code> (-33)	Boolean conditional expression string is invalid.

**Returns**

int

**Syntax**

`AddNewBuildExpr(exprName, exprCondition)`

**Parameters**

Parameter name	Data Type	Optional	Description
<code>exprName</code>	string	No	The name of the Boolean conditional expression.
<code>exprCondition</code>	string	No	The Boolean conditional expression string.

**DeleteBuildExpr****Description**

Deletes a Boolean conditional expression from the document.

The method returns `FE_Success`, on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_ReadOnly</code> (-6)	Document is read-only.
<code>Constants.FE_BadName</code> (-33)	Boolean conditional expression string is invalid.

**Returns**

int

**Syntax**

DeleteBuildExpr (exprName)

**Parameters**

Parameter name	Data Type	Optional	Description
exprName	string	No	The name of the Boolean conditional expression to delete.

**SetActiveBuildExpr****Description**

Applies the Boolean conditional expression to the document.

The method returns `FE_Success`, on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_ReadOnly</code> (-6)	Document is read-only.
<code>Constants.FE_BadName</code> (-33)	Boolean conditional expression string is invalid.

**Returns**

int

**Syntax**

SetActiveBuildExpr (exprName)

## Parameters

Parameter name	Data Type	Optional	Description
exprName	string	No	The name of the Boolean conditional expression to be applied.

## GetActiveBuildExpr

### Description

Returns the name of the active expression in the document or null if none is active.

On failure, the method returns the following value to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDoc Id (-2)</code>	Invalid document.

### Returns

string

### Syntax

```
GetActiveBuildExpr()
```

## GetBuildExpr

### Description

Returns the Boolean conditional expression in the document with the given name or null if none is found.

On failure, the method returns the following value to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDoc Id (-2)</code>	Invalid document.
<code>Constants.FE_BadName ()</code>	The parameter specified by <code>exprName</code> is invalid.

### Returns

string

## Syntax

```
GetBuildExpr (exprName)
```

## Parameters

Parameter name	Data Type	Optional	Description
exprName	string	No	The name of Boolean conditional expression.

## GetBuildExprCatalog

### Description

Returns an array of all Boolean conditional expression names in the document.

On failure, the method returns the following value to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDoc</code> <code>Id (-2)</code>	Invalid document.

### Returns

*Strings*

### Syntax

```
GetBuildExprCatalog()
```

## GetProps

### Description

The `GetProps()` method retrieves the properties of the Doc object.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the Doc object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the Doc object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Element

### GetAttributes

#### Description

Gets an elements attributes.

The `Attributes` structure returned by `GetAttributes()` includes the attributes in the following order:

- Attributes defined in the element definition in the same order in which they are defined in the element definition.
- Undefined attributes in random order.

If an element does not have attributes, the `len` field of the `Attributes` structure is set to 0 and the `val` field is set to `NULL`.

Query the `valflags` field of the `Attributes` structure to determine whether the attribute is valid, or if there are any validation errors. The validation error flags are as follows:

Error	Reason
<code>Constants.FV_AV_REQUIRED (1)</code>	This attribute is required, but it has no value assigned to it.
<code>Constants.FV_AV_INVALID_CHOICE (2)</code>	At least one value for the attribute is not one of the allowed choices.
<code>Constants.FV_AV_INVALID_FORMAT (3)</code>	The attribute value is of the wrong type for the attribute.
<code>Constants.FV_AV_IDREF_UNRESOLVED (4)</code>	The attribute refers to an undefined ID value.
<code>Constants.FV_AV_ID_DUPLICATE_IN_DOC (5)</code>	The attribute value should be unique, but is not unique within the document.
<code>Constants.FV_AV_TOO_MANY_TOKENS (7)</code>	The attribute value has more than one token, but the attribute definition only allows one token.
<code>Constants.FV_AV_UNDEFINED (8)</code>	The attribute is not defined for the containing element.
<code>Constants.FV_AV_OUT_OF_RANGE (9)</code>	The attribute value is out of the range specified in the attribute definition.

On failure, `GetAttributes()` assigns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_WrongProduct (-60)</code>	Current product interface is not structured FrameMaker.
<code>Constants.FE_BadObjId (-3)</code>	Invalid object.

**Returns***Attributes***Syntax**`GetAttributes()`**Delete****Description**

Deletes an element.

See *Delete* under the AFrame class for more information.

**Returns**`int`**Syntax**`Delete()`**UpdateDITAResferences****GetText****Description**

Gets the text from the element.

See *GetText* under the Cell class for more information.

**Returns***TextItems***Syntax**`GetText(flags)`**Parameters**

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.



## GetProps

### Description

Retrieves the properties of the element object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the element object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the element object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

ObjectValid()

## ElementCatalogEntries

An Array of *ElementCatalogEntry* objects with integer indexing and a *length* property.

**concat****Description**

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

**Returns***ElementCatalogEntries***Syntax**`concat (value)`**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>Element Catalog Entries</i>	No	Can also be arrays.

**pop****Description**

Removes the last element from the array.

**Returns***ElementCatalogEntry*

**Syntax**

```
pop ()
```

**push****Description**

Returns the new length of the array.

**Returns**

number

**Syntax**

```
push (value)
```

**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>Element Catalog Entry</i>	No	The value to be pushed.

**ElementCatalogEntry****ElementCatalogEntry****Description**

Describes a catalog entry in an Element Catalog in FrameMaker.

**Returns**

*ElementCatalogEntry*

**Syntax**

```
ElementCatalogEntry(obj, flags)
```

## Parameters

Parameter name	Data Type	Optional	Description
obj	<i>Element Def</i>	No	Element definition.
flags	uint	No	Validation type.

The value of the flags field can be one of the following constants:

Flag	Meaning
Constants.FV_STRICTLY_VALID (0x01)	Catalog entry is strictly valid.
Constants.FV_LOOSELY_VALID (0x02)	Catalog entry is loosely valid.
Constants.FV_ALTERNATIVE (0x04)	Catalog entry is an alternative.
Constants.FV_INCLUSION (0x08)	Catalog entry is valid because it is an inclusion.

If no flags are set, the element is invalid at its current position.

## ElementDef

### NewTextFmtRules

#### Description

Creates a `TextFmtRules` object. The method allows you to associate the created object with a specified property of the parent object. For example, you can associate the text format rule as the text format rule of the element definition.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum (-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

## Returns

*FmtRule*

## Syntax

```
NewTextFmtRules()
```

## NewObjectFmtRules

### Description

Creates a `ObjectFmtRules` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	One

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum (-4)	The property number is invalid.

Error	Reason
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

**Returns***FmtRule***Syntax**

NewObjectFmtRules()

**NewPrefixRules****Description**Creates a `PrefixRules` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

**Returns***FmtRule***Syntax**

NewPrefixRules()

## NewSuffixRules

### Description

Creates a `SuffixRules` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum (-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

### Returns

*FmtRule*

### Syntax

```
NewSuffixRules()
```

## NewFirstPgfrRules

### Description

Creates a `FirstPgfrRules` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

### Returns

*FmtRule*

### Syntax

```
NewFirstPgfRules()
```

### NewLastPgfRules

#### Description

Creates a `LastPgfRules` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.



Error	Reason
Constants.FE_BadPropNum (-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

**Returns***FmtRule***Syntax**`NewLastPgfRules()`**ElementDefIsText****Description**

Some structural elements in FrameMaker documents are placeholders for text. For example, when a Para element contains text with a cross-reference element embedded in it, the ranges of text that surround the cross-reference element are treated as elements themselves. These elements are called `text nodes`.

The `ElementDefIsText()` method checks the value of an element definition to determine whether the element it is applied to is a text node.

The method returns 1 if the element definition corresponds to that of a text node, or 0 if it does not correspond to a text node.

**Returns**`int`**Syntax**`ElementDefIsText()`**NewElement****Description**

Creates a structural element (`FO_Element`) in a FrameMaker document.

`NewElement()` inserts the new element at the specified location in text, using the specified element definition.

For object (noncontainer) elements, `NewElement()` inserts the appropriate type of object for the element. If there is a matching format rule, `NewElement()` uses it to format the object. Otherwise, the method uses one of the following default formats:

Object Type	Object Inserted	Format Used By Default
<code>Constants.FV_FO_XREF</code> (3)	Cross Reference	Undefined XRef
<code>Constants.FV_FO_EQN</code> (7)	Equation	Medium
<code>Constants.FV_FO_MARKER</code> (4)	Marker	Type 11
<code>Constants.FV_FO_TBL</code> (8)	Table with the format specified by the table format.	Format A if it exists; otherwise, a table with a heading row, 8 body rows, a footing row, and 5 columns.
<code>Constants.FV_FO_SYS_VAR</code> (2)	Variable	Filename (Long)
<code>Constants.FV_FO_GRAPHIC</code> (6)	A centered 1.0-inch by 1.0-inch anchored frame below the current position; cropped is off, and floating is on.	-

The method returns the created element, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadNew</code> (-23)	Object cannot be created.
<code>Constants.FE_BadObjId</code> (-3)	Invalid object.
<code>Constants.FE_NotTextObject</code> (-41)	Object specified for the text location is not a paragraph ( <code>FO_Pgf</code> ) or a text line ( <code>FO_TextLine</code> ).
<code>Constants.FE_OffsetNotFound</code> (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.

## Returns

*Element*

## Syntax

`NewElement(textLoc)`

## Parameters

Parameter name	Data Type	Optional	Description
textLoc	<i>TextLoc</i>	No	The text location at which to insert the new element.

## NewElementInHierarchy

### Description

Creates a structural element (`FO_Element`) at a specified location in the element hierarchy of a structured FrameMaker document or book.

To create the root element for a book, you must use `NewElementInHierarchy()`. However, you cannot use `NewElementInHierarchy()` to add elements to an unstructured document. You must structure the document first by adding a root element with `NewElement()`.

For object (noncontainer) elements, `NewElementInHierarchy()` inserts the appropriate type of object for the element. If there is a matching format rule, `NewElementInHierarchy()` uses it to format the object. Otherwise, the method uses one of the following default formats:

Object Type	Object Inserted	Format Used By Default
<code>Constants.FV_FO_XREF</code> (3)	Cross Reference	Undefined XRef
<code>Constants.FV_FO_EQN</code> (7)	Equation	Medium
<code>Constants.FV_FO_MARKER</code> (4)	Marker	Type 11
<code>Constants.FV_FO_TBL</code> (8)	Table with the format and number of rows and columns specified by the table format.	Format A if it exists; otherwise, a table with a heading row, 8 body rows, a footing row, and 5 columns.
<code>Constants.FV_FO_SYS_VAR</code> (2)	Variable	Filename (Long)

Object Type	Object Inserted	Format Used By Default
Constants.FV_FO_GRA PHIC (6)	A centered 1.0-inch by 1.0-inch anchored frame below the current position; cropped is off, and floating is on.	-

The method returns the created element, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadBookId (-66)	Invalid book.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadInsertPos (-65)	<code>elemLoc</code> specifies an invalid place to insert the element; for example, it specifies a position before the highest element in the flow.

### Returns

*Element*

### Syntax

`NewElementInHierarchy(elemLoc)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>elemLoc</code>	<i>Element Loc</i>	No	The location at which the element is to be inserted.

## WrapElement

### Description

Inserts a structural element around the selected text and structural elements in a document.

If the flow that contains the selection is unstructured and the selection does not include the entire flow contents, `WrapElement()` wraps the flow contents into a `NoName` element before wrapping the selection into the specified element definition.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current product interface is not Structured FrameMaker.
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadElementDefId (-28)</code>	Specified element definition is invalid.
<code>Constants.FE_BadSelectionForOperation (-59)</code>	Current text selection is invalid for this operation.

### Returns

Void

### Syntax

`WrapElement()`

### GetAttributeDefs

#### Description

Gets the attribute definitions of an element definition.

On success, the method returns an `AttributeDefs` structure containing the attribute definitions.

The `attrType` field of the returned `AttributeDefs` structure identifies the attribute value's type. It can specify one of the following constants:

Constant	Value
<code>Constant.FV_AT_STRING (0)</code>	Any arbitrary text string.
<code>Constant.FV_AT_STRINGS (1)</code>	One or more arbitrary text strings.
<code>Constants.FV_AT_CHOICES (2)</code>	A value from a list of choices.

Constant	Value
Constants.FV_AT_INTEGER (3)	A signed whole number (optionally restricted to a range of values).
Constants.FV_AT_INTEGER S (4)	One or more integers (optionally restricted to a range of values).
Constants.FV_AT_REAL (5)	A real number (optionally restricted to a range of values).
Constants.FV_AT_REALS (6)	One or more real numbers (optionally restricted to a range of values).
Constants.FV_AT_UNIQUE_ID (7)	A string that uniquely identifies the element.
Constants.FV_AT_UNIQUE_IDREF (8)	A reference to a UniqueID attribute.
Constants.FV_AT_UNIQUE_IDREFS (9)	One or more references to UniqueID attributes.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	Current product interface is not Structured FrameMaker.
Constants.FE_BadObjId (-3)	Invalid object.

## Returns

*AttributeDefs*

## Syntax

```
GetAttributeDefs ()
```

## Delete

### Description

Deletes an element definition.

See [Delete](#) under the AFrame class for more information.

**Returns**`int`**Syntax**`Delete()`**GetProps****Description**

The `GetProps()` method retrieves the properties of an element definition.

See [GetProps](#) under the `AFrame` class for more information.

**Returns***PropVals***Syntax**`GetProps()`**SetProps****Description**

The `SetProps()` method sets the properties of an element definition.

See [SetProps](#) under the `AFrame` class for more information.

**Returns**`Void`**Syntax**`SetProps(setVal)`**Parameters**

Parameter name	Data Type	Optional	Description
<code>setVal</code>	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the element definition is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## ElementLoc

### ElementLoc

### Description

Contains a structure specifying an element location. The ElementLoc structure contains the returned parent element, child element and the offset within the child or parent element.

### Returns

[ElementLoc](#)

### Syntax

```
ElementLoc(parent, child, offset)
```

### Parameters

Parameter name	Data Type	Optional	Description
parent	<a href="#">Element</a>	No	Parent element.
child	<a href="#">Element</a>	No	Child element.
offset	int	No	Offset within the child or parent element.



## ElementRange

### ElementRange

#### Description

A structure specifying an element's range.

#### Returns

*ElementRange*

#### Syntax

ElementRange (beg, end)

#### Parameters

Parameter name	Data Type	Optional	Description
beg	<i>Element Loc</i>	No	Beginning of the element's range.
end	<i>Element Loc</i>	No	End of the element's range.

## Ellipse

### Delete

#### Description

Deletes an ellipse.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

Delete()

## GetProps

### Description

Retrieves the properties of an ellipse.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of an ellipse.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the ellipse is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

ObjectValid()

**FilterArgs****FilterArgs****Description**

FilterArgs stores information that is used in the notification for a file-to-file filter (FA\_Note\_FilterFileToFile). The notification handler receives a pointer to this structure.

**Returns***FilterArgs***Definition**

```
FilterArgs{infile, outfile, cname, informat, outformat, binname, args}
```

**Parameters**

Parameter name	Data Type	Optional	Description
infile	string	No	The file to filter.
outfile	string	No	The filter output file.
cname	string	No	The registered name of the filter.
informat	string	No	The registered incoming format.
outformat	string	No	The registered output format.
binname	string	No	The name of the filter client file.
args	<i>Strings</i>	No	Reserved for future use. Currently NULL.

# Flow

## Delete

### Description

Deletes a flow.

See [Delete](#) under the AFrame class for more information.

### Returns

`int`

### Syntax

```
Delete()
```

## GetText

### Description

Gets the text from the flow.

See [GetText](#) under the Cell class for more information.

### Returns

[TextItems](#)

### Syntax

```
GetText(flags)
```

### Parameters

Parameter name	Data Type	Optional	Description
flags	<code>int</code>	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

Retrieves the properties of the flow.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the flow.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the flow is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## FmtChangeList

### Delete

#### Description

Deletes a format change list.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

### GetProps

#### Description

Retrieves the properties of the format change list.

See [GetProps](#) under the AFrame class for more information.

#### Returns

*PropVals*

#### Syntax

```
GetProps()
```

### SetProps

#### Description

Sets the properties of a format change list.

See [SetProps](#) under the AFrame class for more information.

#### Returns

Void

#### Syntax

```
SetProps(setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the format change list is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## FmtRule

### NewFmtRuleClauses

#### Description

Creates a `FmtRuleClauses` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
FmtRule	FmtRuleClause	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.

Error	Reason
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum (-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

**Returns**

*FmtRuleClause*

**Syntax**

```
NewFmtRuleClauses()
```

**Delete****Description**

Deletes a format rule.

See [Delete](#) under the AFrame class for more information.

**Returns**

```
int
```

**Syntax**

```
Delete()
```

**GetProps****Description**

Retrieves the properties of a format rule.

See [GetProps](#) under the AFrame class for more information.

**Returns**

*PropVals*

**Syntax**

```
GetProps()
```



## SetProps

### Description

Sets the properties of a format rule.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the format rule is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## FmtRuleClause

### NewSubFmtRule

#### Description

Creates a `SubFmtRule` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
FmtRuleClause	FmtRule	One

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

## Returns

*FmtRule*

## Syntax

```
NewSubFmtRule()
```

## NewFmtChangeList

### Description

Creates a `FmtChangeList` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
FmtRuleClause	FmtChangeList	One

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum (-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

**Returns**

*FmtChangeList*

**Syntax**

```
NewFmtChangeList ()
```

**Delete****Description**

Deletes a format rule clause.

See [Delete](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

```
Delete ()
```

**GetProps****Description**

Retrieves the properties of a format rule clause.

See [GetProps](#) under the AFrame class for more information.

**Returns**

*PropVals*

**Syntax**

```
GetProps ()
```

**SetProps****Description**

Sets the properties of a format rule clause.

See [SetProps](#) under the AFrame class for more information.

**Returns**

```
Void
```

**Syntax**

```
SetProps (setVal)
```

**Parameters**

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

**ObjectValid****Description**

Returns 1 if the format rule clause is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

**Returns**

```
int
```

**Syntax**

```
ObjectValid()
```

# Fn

## Delete

### Description

Deletes a footnote.

See [Delete](#) under the AFrame class for more information.

### Returns

`int`

### Syntax

```
Delete()
```

## GetText

### Description

Gets the text from the footnote.

See [GetText](#) under the Cell class for more information.

### Returns

*TextItems*

### Syntax

```
GetText(flags)
```

### Parameters

Parameter name	Data Type	Optional	Description
flags	<code>int</code>	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

Retrieves the properties of a footnote.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of a footnote.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the footnote is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

# Font

## Font

### Description

Font specifies a combination of font characteristics. Each field specifies an index into a list of names in the FrameMaker product session. For example, the `family` field specifies the index of a name in the list of names specified by the session property `FontFamilyNames`.

The `weight` field specifies the index of a name in the list of names specified by the session property `FontWeightNames`.

### Returns

*Font*

### Syntax

```
Font(family, variation, weight, angle)
```

### Parameters

Parameter name	Data Type	Optional	Description
family	uint	No	Index of the font family.
variation	uint	No	Index of the font variation.
weight	uint	No	Index of the font weight.
angle	uint	No	Index of the font angle.

## Fonts

An Array of *Font* objects with integer indexing and a *length* property.

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array as the result of concatenation, with the given values to the end of the original array.

## Returns

*Fonts*

## Syntax

concat (value)

## Parameters

Parameter name	Data Type	Optional	Description
value	<i>Fonts</i>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

*Font*

### Syntax

pop ()

## push

### Description

Pushes the font on the array and the returns the new length of the array.

### Returns

number

### Syntax

push (value)



## Parameters

Parameter name	Data Type	Optional	Description
value	<i>Font</i>	No	Can also be arrays

## Group

A Group object represents a set of grouped objects. Group objects have the following methods.

### Delete

#### Description

Deletes a group.

See [Delete](#) under the AFrame class for more information.

#### Returns

`int`

#### Syntax

```
Delete()
```

### GetProps

#### Description

Retrieves the properties of the group.

See [GetProps](#) under the AFrame class for more information.

#### Returns

*PropVals*

#### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the group.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the group is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## HiddenPage

### GetProps

### Description

Retrieves the properties of the hidden page.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the hidden page.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the hidden page is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## ApplyPageLayout

### Description

Applies the layout of a page to another page.

PageObject is any Page object i.e., *BodyPage*, *MasterPage*, *HiddenPage*.

The method returns `FE_Success` on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_BadOperation (-27)</code>	Parameter specified an invalid operation.
<code>Constants.FE_BadParameter (-43)</code>	Parameter has an invalid value.
<code>Constants.FE_SystemError (-42)</code>	System Error.

### Returns

`int`

### Syntax

`ApplyPageLayout (srcPage)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>srcPage</code>	<code>PageObject</code>	No	The page to which the layout must be applied.

# Inset

## GetIntByName

### Description

Queries an integer facet.

`GetIntByName()` uses a transaction model to query facets. After you have finished a series of queries, you must commit the transaction by calling `GetIntByName()` to query a facet named " ".

If `GetIntByName()` returns 0, it may not indicate an error, because some facet values can be 0. To determine if a returned 0 is a property value or an error, check `FA_errno`.

Returns the specified facet value, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_BadObjId</code> (-3)	Invalid object.
<code>Constants.FE_BadPropNum</code> (-4)	Specified property name is invalid.
<code>Constants.FE_BadPropType</code> e (-5)	Incorrect property type.
<code>Constants.FE_WrongProduct</code> ct (-60)	Current FrameMaker does not support this operation.

### Returns

int

### Syntax

`GetIntByName(propName)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>propName</code>	string	No	The name of the facet to query.

## GetMetricByName

### Description

Queries a metric facet.

`GetMetricByName()` uses a transaction model to query facets. After you have finished a series of queries, you must commit the transaction by calling `GetIntByName()` to query a facet named " ".

If `GetMetricByName()` returns 0, it may not indicate an error, because some facet values can be 0. To determine if a returned 0 is a property value or an error, check `FA_errno`.

Returns the specified facet value, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_BadObjId</code> (-3)	Invalid object.
<code>Constants.FE_BadPropNum</code> (-4)	Specified property name is invalid.
<code>Constants.FE_BadPropType</code> (-5)	Incorrect property type.
<code>Constants.FE_WrongProduct</code> (-60)	Current FrameMaker does not support this operation.

### Returns

int

### Syntax

`GetMetricByName(propName)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>propName</code>	string	No	The name of the facet to query.

## GetUBytesByName

### Description

Queries an unsigned bytes inset facet. The standard facets, EPSI and FrameImage, are examples of unsigned bytes facets.

If a facet contains a large amount of data, `GetUBytesByName()` only gets a portion of the data each time you call it.

To query a facet that contains a large amount of data, call `GetUBytesByName()` repeatedly until you have retrieved all the data (that is, until `UBytes.len` is 0).

`GetUBytesByName()` uses a transaction model to query facets. After you have finished a series of queries, you must commit the transaction by calling `GetIntByName()` to query a facet named " `".

On success, the method returns a `UBytes` object containing a portion of the facet's data.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_BadObjId</code> (-3)	Invalid object.
<code>Constants.FE_BadPropNum</code> (-4)	Specified property name is invalid.
<code>Constants.FE_BadPropType</code> (-5)	Incorrect property type.
<code>Constants.FE_WrongProduct</code> (-60)	Current FrameMaker does not support this operation.

**NOTE:** The returned `UBytes` object references memory that is allocated by the method. Use `DeallocateUBytes()` to free this memory when you are done using it.

### Returns

`UBytes`

### Syntax

```
GetUBytesByName(propName)
```

## Parameters

Parameter name	Data Type	Optional	Description
propName	string	No	The name of the facet to query.

## SetIntByName

### Description

Sets an integer inset facet.

`SetIntByName()` uses a transaction model to set facets. After you have finished setting facets, you must commit the transaction by calling `SetIntByName()` to set a facet named " " to 0.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_BadObjId</code> (-3)	Invalid object.
<code>Constants.FE_BadPropNum</code> (-4)	Specified property name is invalid.
<code>Constants.FE_BadName</code> (-33)	Specified name is illegal.
<code>Constants.FE_WrongProduct</code> (-60)	Current FrameMaker does not support this operation.

### Returns

Void

### Syntax

```
SetIntByName(propName, setVal)
```



## Parameters

Parameter name	Data Type	Optional	Description
propName	string	No	The name of the facet to set.
setVal	int	No	The value to which the facet must be set.

## SetMetricByName

### Description

Sets a metric facet.

`SetMetricByName()` uses a transaction model to set facets. After you have finished setting facets, you must commit the transaction by calling `SetIntByName()` to set a facet named "" to 0.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_BadName</code> (-33)	Specified name is illegal.
<code>Constants.FE_BadObjId</code> (-3)	Invalid object.
<code>Constants.FE_BadPropType</code> (-5)	Incorrect property type for this function.
<code>Constants.FE_OutOfRange</code> (-7)	Specified property value is out of the legal range for the specified .property.
<code>Constants.FE_ReadOnly</code> (-6)	Property is read-only.
<code>Constants.FE_WrongProduct</code> (-60)	Current FrameMaker does not support this operation.

### Returns

Void

## Syntax

```
SetMetricByName(propName, setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
propName	string	No	The name of the facet to set.
setVal	int	No	The value to which the facet must be set.

## SetUBytesByName

### Description

Sets an unsigned bytes inset facet. The standard facets, EPSI and FramelImage, are examples of unsigned bytes facets.

To set a UBytes facet:

- 1) Call `SetUBytesByName()` to set the facet data.

*If you are setting a facet with less than 10K of data, you need to call `SetUBytesByName()` only once.*

*If you are setting the facet with more than 10K of data, you should call `SetUBytesByName()` multiple times, setting a chunk of the data each time. You can size the chunks between 0 and 10K.*

**NOTE:** *If you use larger chunks, the set operation goes faster. However, if the chunks are too large, you may overload your platform's interapplication communication mechanism.*

- 2) Call `SetUBytesByName()` with `propNameset` to an empty string (" ").

*This lets the ESTK know that you have finished setting up the facet.*

The method does not return anything on success.

On failure, the method returns one of the following value to `FA_errno`:

Error	Reason
<code>Constants.FE_Transport (-1)</code>	A transportation error occurred.

### Returns

Void

### Syntax

```
SetUBytesByName(propName, setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
propName	string	No	The name of the facet to set.
setVal	<i>UBytes</i>	No	The value to which the facet must be set.

## Delete

### Description

Deletes an inset.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## DeletePropByName

### Description

Deletes an inset facet. Except for inset facets, all properties are identified by integer constants. Facets are the only properties identified by name.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadPropNum</code> (-4)	Specified property number is invalid.
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_BadObjId</code> (-3)	Invalid object.
<code>Constants.FE_BadPropType</code> e (-5)	Incorrect property type for this function.

## Returns

Void

## Syntax

```
DeletePropByName (propName)
```

## Parameters

Parameter name	Data Type	Optional	Description
propName	string	No	The name of the facet to be deleted.

## GetProps

### Description

Retrieves the properties of the inset.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the inset.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the inset is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Ints

An Array of objects with integer indexing and a *length* property.

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

*Ints*

#### Syntax

```
concat(value)
```

**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>Ints</i>	No	Can also be arrays.

**pop****Description**

Removes the last element from the array.

**Returns**

`int`

**Syntax**

`pop()`

**push****Description**

Returns the new length of the array.

**Returns**

`number`

**Syntax**

`push(value)`

**Parameters**

Parameter name	Data Type	Optional	Description
value	<code>int</code>	No	Can also be arrays.

# Iterator

## GetNextIteratorItem

### Description

Retrieves the next iterator in the document.

### Returns

*PropVals*

### Syntax

```
GetNextIteratorItem()
```

## Delete

### Description

Deletes an iterator.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the iterator.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the iterator.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the iterator is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## KeyCatalog

### NewKeyDefinition

#### Description

Adds a new key definition to the specified key catalog.



**Returns**

void

**Syntax**

```
NewKeyDefinition(key, href, srcType, srcFile, flags);
```

**Parameters**

Parameter name	Data Type	Optional	Description
key	int	No	The tag of the key for which the key definition is being added.
href	int	No	The complete path of the file that the key refers to.
srcType	int	No	The type of the file that contains the key definition. See the table below for a list of values.
srcFile	int	No	The complete path of the file that contains the key definition.
flags	int	No	Bit flags specifying information about the key definition. See the table below for a list of flags.

srcType can have one of the following values:

srcType	Meaning
Constants.FV_KeySrcTypeNone	Source file type not specified.
Constants.FV_KeySrcTypeDitamap	Source file is a DITA Map.

You can OR the following bit-flags into flags:

Bit mask	Meaning
Constants.FF_DUPLICATE_KEY_DEFINITION	The specified key definition is duplicate (that is, it already exists in the Key Catalog) and will not be used as active definition for resolving keys.

Bit mask	Meaning
Constants.FF_FOUND_IN_REFERENCED_FILE	The specified key definition is contained in a file referenced directly or indirectly from the file that contains the key definition (srcFile).
Constants.FF_INVALID_KEY	The specified key definition is invalid due to some reason but will still be kept in the Key Catalog.

If NewKeyDefinition() fails, the method assigns following values to FA\_erno:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.
Constants.FE_BadKey	The Key provided is not valid.
Constants.FE_KeyDefinitionAlreadyExists	The definition for the specified key is already available in the Key Catalog and the key definition provided is not duplicate.

## UpdateKeyDefinition

### Description

Updates the specified key definition field for the specified key in the specified key catalog.

### Returns

void

### Syntax

```
UpdateKeyDefinition(key, keyField, valuep);
```

### Parameters

Parameter name	Data Type	Optional	Description
key	int	No	The tag of the key for which the key definition is being updated.
keyField	int	No	The key field (or key information) that is being updated.
valuep	int	No	The value to update the keyField to.

The valid keyField values and the corresponding value type are as follows:

<b>keyField</b>	<b>Value type</b>
Constants.FV_KeydefKeyTarget	FT_String
Constants.FV_KeydefKeySrcFile	FT_String
Constants.FV_KeydefKeySrcType	FT_Integer
Constants.FV_KeydefKeyVarList	FT_Vals
Constants.FV_KeydefKeyDefaultText	FT_String
Constants.FV_KeydefKeyFoundInRefFile	FT_Integer
Constants.FV_KeydefKeyInvalid	FT_Integer
Constants.FV_KeydefKeyAttrs	FT_AttributesEx

If UpdateKeyDefinition() fails, the method assigns following values to FA\_errno:

<b>Error</b>	<b>Reason</b>
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.
Constants.FE_BadKey	The Key provided is not valid.
Constants.FE_KeyDefinitionDoesNotExist	The definition for the specified key is not available in the Key Catalog.
Constants.FE_BadValue	The value is either not specified or it is not as expected for the specified 'keyField'.
Constants.FE_ReadOnly	(only for keyField=FV_KeydefKeyTag or FV_KeydefKeyDuplicate). The key field cannot be changed/updated.
Constants.FE_InvAttribute	(only for keyField=FV_KeydefKeyAttrs) The Attribute information provided is not valid.
Constants.FE_WrongProduct	(only for keyField=FV_KeydefKeyAttrs) Current FrameMaker product doesn't support the operation.
Constants.FE_BadKeyField	The key field provided is not valid.

## GetKeyDefinition

### Description

Gets the specified key definition field for the specified key from the specified key catalog.

### Returns

TypedVal

### Syntax

```
GetKeyDefinition(key, keyField);
```

### Parameters

Parameter name	Data Type	Optional	Description
key	int	No	The tag of the key for which the key definition is being asked for.
keyField	int	No	The key field (or key information) that is being asked for.

The valid keyField values and the corresponding value type are as follows:

keyField	Value type
Constants.FV_KeydefKeyTag	FT_String
Constants.FV_KeydefKeyTarget	FT_String
Constants.FV_KeydefKeySrcFile	FT_String
Constants.FV_KeydefKeyDuplicate	FT_Integer
Constants.FV_KeydefKeySrcType	FT_Integer
Constants.FV_KeydefKeyVarList	FT_Vals
Constants.FV_KeydefKeyDefaultText	FT_String

keyField	Value type
Constants.FV_KeydefKeyFoun dInRefFile	FT_Integer
Constants.FV_KeydefKeyInVal id	FT_Integer
Constants.FV_KeydefKeyAttrs	FT_AttributesEx

If GetKeyDefinition() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.
Constants.FE_BadKey	The Key provided is not valid.
Constants.FE_KeyCatalogNot Loaded	The Key Catalog provided is currently not loaded.
Constants.FE_KeyCatalogIsSt ale	The Key Catalog provided is currently marked as stale and needs to be re-loaded before using.
Constants.FE_KeyDefinitionD oesNotExist	The definition for the specified key is not available in the Key Catalog.
Constants.FE_WrongProduct	(only for keyField="FV_KeydefKeyAttrs") Current FrameMaker product doesn't support the operation.
Constants.FE_BadKeyField	The key field provided is not valid.

## GetAllKeyDefinitions

### Description

Gets all the key definitions from the specified key catalog.

### Returns

TypedVals

### Syntax

```
GetAllKeyDefinitions(filterType);
```

**Parameters**

Parameter name	Data Type	Optional	Description
filterType	int	No	Specifies the kind of key fields to get for each key definition. filterType can have the following values: FV_KeyDefFieldsTypePrimary: Get only the primary key fields (Tag, Target, SrcFile, and Duplicate) FV_KeyDefFieldsTypeAll: Get all key fields.

Returns the information in a TypedVals structure as follows: FieldTag is of type FT\_Integer. FieldValue is of type as specified in the table below.

FieldTag value	FieldValue type
Constants.FV_KeydefKeyAttrs	FT_AttributesEx
Constants.FV_KeydefKeyDefaultText	FT_String
Constants.FV_KeydefKeyDuplicate	FT_Integer
Constants.FV_KeydefKeyFoundInRefFile	FT_Integer
Constants.FV_KeydefKeyInvalid	FT_Integer
Constants.FV_KeydefKeySrcFile	FT_String
Constants.FV_KeydefKeySrcType	FT_Integer
Constants.FV_KeydefKeyTag	FT_String
Constants.FV_KeydefKeyTarget	FT_String
Constants.FV_KeydefKeyVarList	FT_Vals

If GetAllKeyDefinitions() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.
Constants.FE_KeyCatalogNotLoaded	The Key Catalog provided is currently not loaded.
Constants.FE_KeyCatalogIsStale	The Key Catalog provided is currently marked as stale and needs to be re-loaded before using.
Constants.FE_BadFilterType	The filter type provided is not valid.

## DeleteAllKeyDefinitions

### Description

Deletes all the key definitions in the specified key catalog.

### Returns

void

### Syntax

```
DeleteAllKeyDefinitions();
```

If DeleteAllKeyDefinitions() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.

## GetAllKeys

### Description

Gets all the key tags from the specified key catalog.

### Returns

Strings

### Syntax

```
GetAllKeys();
```

If GetAllKeys() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.
Constants.FE_KeyCatalogNot Loaded	The Key Catalog provided is currently not loaded.
Constants.FE_KeyCatalogIsStale	The Key Catalog provided is currently marked as stale and needs to be re-loaded before using.

## Delete

### Description

Deletes a key catalog.

See [Delete](#) under the AFrame class for more information.

### Returns

`int`

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the key catalog.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the key catalog.

See [SetProps](#) under the AFrame class for more information.



## Returns

Void

## Syntax

```
SetProps (setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the key catalog is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Line

### Delete

#### Description

Deletes a line.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

**Syntax**

```
Delete()
```

**GetProps****Description**

Retrieves the properties of the line.

See [GetProps](#) under the AFrame class for more information.

**Returns**

*PropVals*

**Syntax**

```
GetProps()
```

**SetProps****Description**

Sets the properties of the line.

See [SetProps](#) under the AFrame class for more information.

**Returns**

Void

**Syntax**

```
SetProps(setVal)
```

**Parameters**

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

**ObjectValid****Description**

Returns 1 if the line is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

```
ObjectValid()
```

## Marker

**Delete****Description**

Deletes a marker.

See [Delete](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

```
Delete()
```

**GetProps****Description**

Retrieves the properties of the marker.

See [GetProps](#) under the AFrame class for more information.

**Returns**

*PropVals*

**Syntax**

```
GetProps()
```

**SetProps****Description**

Sets the properties of the marker.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The properties list.

## ObjectValid

### Description

Returns 1 if the marker is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## MarkerType

### Delete

#### Description

Deletes a marker type.

See [Delete](#) under the AFrame class for more information.

### Returns

int

## Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the marker type.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

## Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the marker type.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

## Syntax

```
SetProps(setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the marker type is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

```
ObjectValid()
```

## MasterPage

**Delete****Description**

Deletes a master page.

See [Delete](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

```
Delete()
```

**GetProps****Description**

Retrieves the properties of the master page.

See [GetProps](#) under the AFrame class for more information.

**Returns**

*PropVals*

**Syntax**

```
GetProps()
```

**SetProps****Description**

Sets the properties of the master page.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the master page is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## ApplyPageLayout

### Description

Copies the layout of one master page to another.

PageObject is any Page object i.e., [BodyPage](#), [MasterPage](#), [HiddenPage](#).

See [ApplyPageLayout](#) under the Body Page class for more information.

### Returns

int

## Syntax

```
ApplyPageLayout (srcPage)
```

## Parameters

Parameter name	Data Type	Optional	Description
srcPage	PageObject	No	The page to which the layout is to be applied.

## Math

### Delete

#### Description

Deletes a math object.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

### Syntax

```
Delete()
```

### GetProps

#### Description

Retrieves the properties of the math object.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

### Syntax

```
GetProps()
```



## SetProps

### Description

Sets the properties of the math object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the math object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## MathML

### Delete

#### Description

Deletes a math object.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the math object.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the math object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the math object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Menu

### DefineAndAddCommand

#### Description

Defines a command (`FO_Command object`) and adds it to a menu or menu bar.

The following table lists some FrameMaker product menus and the names you use to specify them:

Menu Title	Menu Name
Edit	EditMenu
File	FileMenu
Format	FormatMenu
Graphics	GraphicsMenu
Special	SpecialMenu
Table	TableMenu
View	ViewMenu
Help	!HelpMenu

If you call `DefineAndAddCommand()` and specify the name of a command that is already defined in the user's menu configuration files, FrameMaker gives precedence to the definition in the configuration files.

If the configuration files assign a label or a shortcut to the command, then FrameMaker uses it instead of the one you specify. If the command is already a menu item, FrameMaker ignores the menu that you specify and leaves the menu item where it is.

**NOTE:** *If you want to add a command to more than one menu, do not call `DefineAndAddCommand()` repeatedly to add the command to the menus. To add a command to multiple menus, define the command first by calling `DefineCommand()`—or call `iDefineAndAddCommand()`, if you want to define and add the command to a menu at the same time—and then call `AddCommandToMenu()` to add the command to other menus.*

The method returns the command, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specified an invalid operation.
<code>Constants.FE_NotMenu (-72)</code>	The specified menu does not exist.
<code>Constants.FE_BadParameter (-43)</code>	The specified parameter has an invalid value.
<code>Constants.FE_SystemError (-42)</code>	System Error

## Returns

*Command*

## Syntax

```
DefineAndAddCommand(cmdNum, name, label, shortcut)
```

## Parameters

Parameter name	Data Type	Optional	Description
cmdNum	number	No	The integer that the FrameMaker product passes to your client's <code>Command()</code> function when the user chooses the menu item or types the keyboard shortcut for the command. The value must be unique for each command in your client, but need not be unique for different clients.
name	string	No	A unique name for the command. If the user or a client has already defined a command or menu with this name, the new command replaces it.
label	string	No	The title of the command as it appears on the menu.
shortcut	string	No	The keyboard shortcut sequence. Many FrameMaker product commands use shortcuts beginning with Escape ( <code>\!</code> ). To specify Escape when you create a command, use <code>\\!</code> in the string you pass to <code>shortcut</code> .

## AddCommandToMenu

### Description

Adds a FrameMaker product command or a client-defined command to a menu.

`AddCommandToMenu()` adds the command at the bottom of the specified menu. To change a command's position on a menu, set its `PrevMenuItemInMenu` and `NextMenuItemInMenu` properties.

To add a command that you have created, set the `command` parameter to the object returned by the `DefineCommand()` call that created the command. To add a FrameMaker product command, you must get its object. To get its object, call `GetNamedObject()` with the `objectName` parameter set to its name.

The method returns the command, on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker product does not support this operation or fmbatch is running.
Constants.FE_BadOperation (-27)	Parameters specified an invalid operation.
Constants.FE_NotCommand (-73)	The command parameter did not specify a command.
Constants.FE_NotMenu (-72)	The specified menu does not exist.
Constants.FE_BadParameter (-43)	The specified parameter has an invalid value.
Constants.FE_SystemError (-42)	System Error

**Returns**

number

**Syntax**

AddCommandToMenu (command)

**Parameters**

Parameter name	Data Type	Optional	Description
command	<i>Command</i>	No	The command to add.

**AddMenuToMenu****Description**

Adds a FrameMaker product menu or a menu that you have created to another menu or menu bar.

To add a menu to one of FrameMaker menus or menu bars, you must get the menu or menu bar's object. To get its object, call `GetNamedObject()` with the `objectName` parameter set to its name.

**NOTE:** Your menu appears only on the menu bar you specify. For example, if you only add a menu to the `!MakerMainMenu` menu bar, the menu will not appear if the user switches to quick menus. For your menu to appear after the user has switched to quick menus, you must also add it to `!QuickMakerMainMenu`.

The following table lists the types of menus you can add a menu to and how FrameMaker implements the added menu.

Type of menu or menu bar you are adding a menu to	How FrameMaker implements the added menu	FrameMaker product adds the menu
Menu bar	Pull-down menu	At the right of the menu bar.
Pull-down menu	Pull-right menu	At the bottom of the pull-down menu.
Pop-up menu	Pull-right menu	At the bottom of the pop-up menu.
Pull-right menu	Pull-right menu	At the bottom of the pull-right menu.

To change a menu's position on a menu or menu bar after you add it, set its `PrevMenuItemInMenu` and `NextMenuItemInMenu` properties.

The method returns the command, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_NotMenu (-72)</code>	The specified menu does not exist.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specified an invalid operation.
<code>Constants.FE_BadParameter (-43)</code>	The specified parameter has an invalid value.
<code>Constants.FE_SystemError (-42)</code>	System Error

## Returns

`int`

## Syntax

`AddMenuToMenu (menu)`

## Parameters

Parameter name	Data Type	Optional	Description
menu	<i>Menu</i>	No	The new menu to be added.

## DefineAndAddMenu

### Description

Defines a menu (FO\_Menu object) and adds it to another menu.

To add a menu to one of FrameMaker's menu bars, you must get the menu or menu bar's object. To get its object, call `GetNamedObject()` with the `objectName` parameter set to its name.

The following table lists some of the menu bars that you can add menus to and the strings that specify them. Menu bar names preceded by an exclamation mark (!) cannot be removed by the user.

FrameMaker menubar	Menubar string
Menu bar for documents (complete menus)	!MakerMainMenu
Menu bar for documents (quick menus)	!QuickMakerMainMenu
Menu bar for documents (custom menus)	!CustomMakerMainMenu
Menu bar for books (complete menus)	!BookMainMenu
Menu bar for books (quick menus)	!QuickBookMainMenu
Structure menu bar (structured product interface only)	!StructureViewMainMenu
Pop-up menu for documents (complete menus; UNIX only)	!MakerPopup
Pop-up menu for documents (quick menus; UNIX only)	!QuickMakerPopup
View-only menu bar	!ViewOnlyMainMenu
View-only pop-up menu (UNIX only)	!ViewerPopup

**NOTE:** Your menu appears only on the menu bar you specify. For example, if you only add a menu to the `!MakerMainMenu` menu bar, the menu will not appear if the user switches to quick menus. For your menu to appear after the user has switched to quick menus, you must also add it to `!QuickMakerMainMenu`.



If you call `DefineAndAddMenu()` and specify the name of a menu that is already defined in the user's menu configuration files, FrameMaker gives precedence to the definition in the configuration files.

If the configuration files assign a label to the menu, FrameMaker uses it instead of the one you specify. If the menu is already on a menu or menu bar, FrameMaker ignores the menu that you specify and leaves the menu where it is.

The following table lists the types of menus you can add a menu to and how FrameMaker implements the added menu.

Type of menu or menu bar you are adding a menu to	How FrameMaker implements the added menu	FrameMaker product adds the menu
Menu bar	Pull-down menu	At the right of the menu bar.
Pull-down menu	Pull-right menu	At the bottom of the pull-down menu.
Pop-up menu	Pull-right menu	At the bottom of the pop-up menu.
Pull-right menu	Pull-right menu	At the bottom of the pull-right menu.

The method returns the menu, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_NotMenu (-72)</code>	The specified menu does not exist.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specified an invalid operation.
<code>Constants.FE_BadParameter (-43)</code>	The specified parameter has an invalid value.
<code>Constants.FE_SystemError (-42)</code>	System Error

## Returns

*Menu*

## Syntax

```
DefineAndAddMenu(name, label)
```

## Parameters

Parameter name	Data Type	Optional	Description
name	string	No	A unique name for the new menu. If the user or an ESTK client has already defined a command or menu with this name, the new menu replaces it.
label	string	No	The title of the menu as it appears on the menu bar or menu.

## MenuItemInMenu

### Description

Determines if a menu item or menu is on a menu or menu bar.

On success, the method returns the menu on which the menu item is found.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specified an invalid operation.
<code>Constants.FE_BadParameter (-43)</code>	The specified parameter has an invalid value.

### Returns

*Menu*

### Syntax

```
MenuItemInMenu(menuitem, recursive)
```

## Parameters

Parameter name	Data Type	Optional	Description
menuItem	MenuItemComponent	No	The menu item or menu to find.
recursive	bool	No	Specifies whether to search the submenus of the menu or menu bar specified by menuItem. Specify <code>True</code> to search them.

## Delete

### Description

Deletes a menu or menu item.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the menu or menu item.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the menu or menu item.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the menu or menu item is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Delete

### Description

Deletes a menu or menu item.

See [Delete](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

Delete()

## MenuItemSeparator

**GetProps****Description**

Retrieves the properties of the menu item separator.

See [GetProps](#) under the AFrame class for more information.

**Returns**

*PropVals*

**Syntax**

GetProps()

**SetProps****Description**

Sets the properties of the menu item separator.

See [SetProps](#) under the AFrame class for more information.

**Returns**

Void

**Syntax**

SetProps(setVal)

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the menu item separator is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Delete

### Description

Deletes a menu item separator.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## Metrics

An Array of objects with integer indexing and a *length* property.

## concat

### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

### Returns

*Metrics*

### Syntax

```
concat(value)
```

### Parameters

Parameter name	Data Type	Optional	Description
value	<i>Metrics</i>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

int

### Syntax

```
pop()
```

## push

### Description

Returns the new length of the array.

### Returns

number

## Syntax

```
push(value)
```

## Parameters

Parameter name	Data Type	Optional	Description
value	int	No	The value to push into the array.

## Pgf

### Delete

#### Description

Deletes a paragraph.

See [Delete](#) under the AFrame class for more information.

#### Returns

```
int
```

### Syntax

```
Delete()
```

### GetText

#### Description

Gets the text from the paragraph.

See [GetText](#) under the Cell class for more information.

#### Returns

```
TextItems
```

### Syntax

```
GetText(flags)
```



## Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

Retrieves the properties of the paragraph.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the paragraph.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the paragraph is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## PgfFmt

### Delete

#### Description

Deletes a paragraph format.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

## GetProps

#### Description

Retrieves the properties of the paragraph format.

See [GetProps](#) under the AFrame class for more information.

#### Returns

*PropVals*

#### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the paragraph format.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the paragraph format is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

# Point

## Point

### Description

Point describes an individual coordinate pair. FrameMaker measures coordinates from the upper-left corner of the parent frame

### Returns

*Point*

### Syntax

`Point(x, y)`

### Parameters

Parameter name	Data Type	Optional	Description
x	int	No	X coordinate
y	int	No	Y coordinate

## Points

An Array of *Point* objects with integer indexing and a *length* property.

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

*Points*

#### Syntax

`concat(value)`

## Parameters

Parameter name	Data Type	Optional	Description
value	<i>Points</i>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

*Point*

### Syntax

`pop()`

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

`push(value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<i>Point</i>	No	The value to push into the array.

# Polygon

## Delete

### Description

Deletes a polygon.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the polygon.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the polygon.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the polygon is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Polyline

### Delete

#### Description

Deletes a polyline.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

### GetProps

#### Description

Retrieves the properties of the polyline.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the polyline.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the polyline is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```



# PropIdent

## PropIdent

### Description

PropIdent provides a property identifier. Properties can be identified by either a name or a number (integer constant). ESTK provides defined constants for property numbers (for example, Fill and Height). Only inset properties (facets) are identified by names. If a property is identified by a name, `PropIdent.num` is set to 0. If a property is identified by a number, `PropIdent.name` is set to a null string.

### Returns

*PropIdent*

### Syntax

```
PropIdent(num, name)
```

### Parameters

Parameter name	Data Type	Optional	Description
num	int	No	Property number.
name	string	No	Property name.

# PropVal

## PropVal

### Description

Describes a property-value pair.

### Returns

*PropVal*

### Syntax

```
PropVal(propIdent, propVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
propIdent	<i>PropIdent</i>	No	Property identifier.
propVal	<i>TypedVal</i>	No	Property value.

## PropVals

An Array of *PropVal* objects with integer indexing and a *length* property.

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

*PropVals*

#### Syntax

```
concat (value)
```

#### Parameters

Parameter name	Data Type	Optional	Description
value	<i>PropVals</i>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

**Returns***PropVal***Syntax**`pop()`**push****Description**

Returns the new length of the array.

**Returns**

number

**Syntax**`push(value)`**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>PropVal</i>	No	The value to push into the array.

## Rectangle

**Delete****Description**

Deletes a rectangle.

See [Delete](#) under the AFrame class for more information.

**Returns**

int

**Syntax**`Delete()`

## GetProps

### Description

Retrieves the properties of the rectangle.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the rectangle.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the rectangle is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

## Returns

int

## Syntax

```
ObjectValid()
```

# RefPage

## Delete

### Description

Deletes a reference page.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the reference page.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the reference page.

See [SetProps](#) under the AFrame class for more information.

**Returns**

Void

**Syntax**

SetProps (setVal)

**Parameters**

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

**ObjectValid****Description**

Returns 1 if the reference page is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.**Returns**

int

**Syntax**

ObjectValid()

**ApplyPageLayout****Description**PageObject is any Page object i.e., [BodyPage](#), [MasterPage](#), [HiddenPage](#).See [ApplyPageLayout](#) under the BodyPage class for more information.**Returns**

int

**Syntax**

ApplyPageLayout (srcPage)

## Parameters

Parameter name	Data Type	Optional	Description
srcPage	PageObject	No	The page to which the layout is to be applied.

## RoundRect

### Delete

#### Description

Deletes a round rectangle.

See [Delete](#) under the AFrame class for more information.

#### Returns

`int`

#### Syntax

```
Delete()
```

### GetProps

#### Description

Retrieves the properties of the round rectangle.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

#### Syntax

```
GetProps()
```

### SetProps

#### Description

Sets the properties of the rounded rectangle.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the rounded rectangle is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Row

### AddRows

#### Description

Adds one or more rows to a table.

The following table lists the constants you can specify for the `direction` parameter:



Direction	Meaning
Constants.FV_Above (1240)	Add rows above the current reference row.
Constants.FV_Below (1241)	Add rows below the current reference row
Constants.FV_Body (1237)	Add rows at the bottom of the existing body rows.
Constants.FV_Footing (1239)	Add rows at the bottom of the existing footer rows.
Constants.FV_Heading (1238)	Add rows at the bottom of the existing heading rows

The method returns `FE_Success` on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker product does not support this operation.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid row number.
Constants.FE_BadOperation (-27)	Parameters specified an invalid operation.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.

### Returns

`int`

### Syntax

```
AddRows(direction, numNewRows)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>direction</code>	<code>int</code>	No	The direction from the reference row in which to add rows.
<code>numNewRows</code>	<code>int</code>	No	The number of rows to add.

## Delete

### Description

Deletes a row.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the row.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the row.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the row is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Rubi

### Delete

#### Description

Deletes a rubi object.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

## GetText2

### Description

Gets the text from all the objects available to `GetText()`, as well as an extended set of document objects that includes the rubi object.

`TextItems.dataType` can be one of the following constants (in addition to the constants returned by `GetText()`):

Value	Representation	Data
<code>Constants.FTI2_RubiTextBegin (0x80000001)</code>	The beginning of rubi text.	The <code>FO_Rubi</code> object for the rubi composite that contains the rubi text.
<code>Constants.FTI2_RubiTextEnd (0x80000002)</code>	The end of rubi text.	The <code>FO_Rubi</code> object for the rubi composite that contains the rubi text.
<code>Constants.FTI2_RubiCompositeBegin (0x80000004)</code>	The beginning of a rubi composite.	<code>FO_Rubi</code> object.
<code>Constants.FTI2_RubiCompositeEnd (0x80000008)</code>	The end of a rubi composite.	<code>FO_Rubi</code> object.
<code>Constants.FTI2_InlineComponentBegin (0x80000010)</code>	The beginning of inline component.	<i>InlineComponent</i> object.
<code>Constants.FTI2_InlineComponentEnd (0x80000020)</code>	The end of inline component.	<i>InlineComponent</i> object.

See [GetText](#) under the Cell class for more information.

### Returns

*TextItems*

### Syntax

```
GetText2(flags, flags2)
```

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

Parameter name	Data Type	Optional	Description
flags2	int	No	Bit flags for the extended set of text item types that specify the type of text items to retrieve. To get specific types of text items from the extended set, OR the constants that represent them (for example, use a bitwise OR to <code>combineConstants.FTI2_RubiTextBegin (0x80000001) and Constants.FTI2_RubiTextEnd (0x80000002)</code> ) into flags2. To get all types of the extended set of text items, specify -1.

## GetProps

### Description

Retrieves the properties of the rubi object.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the rubi object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the rubi object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## RulingFmt

### Delete

#### Description

Deletes a ruling format.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

### GetProps

#### Description

Retrieves the properties of the ruling format.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the ruling format.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the ruling format is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Strings

An Array of objects with integer indexing and a *length* property.

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

*Strings*

#### Syntax

```
concat(value)
```

#### Parameters

Parameter name	Data Type	Optional	Description
value	<i>Strings</i>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

#### Returns

```
string
```

#### Syntax

```
pop()
```

### push

#### Description

Returns the new length of the array.



**Returns**

number

**Syntax**

`push(value)`

**Parameters**

Parameter name	Data Type	Optional	Description
value	string	No	The value to push into the array.

## SubCol

**Delete****Description**

Deletes a subcol object.

See [Delete](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

`Delete()`

**GetText****Description**

Gets the text from the subcol object.

See [GetText](#) under the Cell class for more information.

**Returns**

*TextItems*

**Syntax**

`GetText(flags)`

## Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

Retrieves the properties of the subcol object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the subcol object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the subcol object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Tab

### Tab

### Description

Describes an individual tab. Note that the character specified by `decimal` must be a single byte character.

The `type` field can contain one of the following constants:

Constant	Tab Type
Constants.FV_TAB_LEFT (0x1)	Left tab.
Constants.FV_TAB_CENTER (0x2)	Center tab.
Constants.FV_TAB_RIGHT (0x3)	Right tab.
Constants.FV_TAB_DECIMAL (0x4).	Decimal tab.
Constants.FV_TAB_RELATIVE_LEFT (0x5)	Relative left tab (allowed only for format change lists).
Constants.FV_TAB_RELATIVE_CENTER (0x6)	Relative center tab (allowed only for format change lists).
Constants.FV_TAB_RELATIVE_RIGHT (0x7)	Relative right tab (allowed only for format change lists).
Constants.FV_TAB_RELATIVE_DECIMAL (0x8)	Relative decimal tab (allowed only for format change lists).

**Returns***Tab***Syntax**`Tab(x, type, leader, decimal)`**Parameters**

Parameter name	Data Type	Optional	Description
x	int	No	Offset from the left margin.
type	uint	No	Type of tab.
leader	string	No	String that appears before the tab.
decimal	uint	No	Character to align tab around - For example " , " .

**Tabs**

An Array of *Tab* objects with integer indexing and a *length* property.

**concat****Description**

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

**Returns***Tabs***Syntax**`concat(value)`

## Parameters

Parameter name	Data Type	Optional	Description
value	<i>Tabs</i>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

*Tab*

### Syntax

```
pop()
```

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

```
push(value)
```

## Parameters

Parameter name	Data Type	Optional	Description
value	<i>Tab</i>	No	

# Tbl

## AddCols

### Description

Adds columns to a table.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	invalid document.
Constants.FE_BadParameter (-43)	A parameter specified is invalid.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadOperation (-27)	The method specified an invalid operation.

### Returns

`int`

### Syntax

```
AddCols(refColNum, direction, numNewCols)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>refColNum</code>	<code>int</code>	No	The column at which to start adding columns. The columns are numbered from left to right starting with column 0.
<code>direction</code>	<code>int</code>	No	The direction from the reference column in which to add columns. To add columns to the left of the reference column, specify <code>FV_Left</code> . To add them to the right, specify <code>FV_Right</code> .

Parameter name	Data Type	Optional	Description
numNewCols	int	No	The number of columns to add.

## DeleteCols

### Description

Deletes columns from a table. To delete an entire table, use `Delete()`.

The method deletes the column specified by `delColNum` and `(numDelCols-1)` columns to the right of it.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the requested operation.
Constants.FE_BadOperation (-27)	The method specified an invalid operation.
Constants.FE_BadDocId (-2)	invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_TableInLockedException (-101)	The table is in a locked text inset.

### Returns

int

### Syntax

```
DeleteCols (delColNum, numDelCols)
```

## Parameters

Parameter name	Data Type	Optional	Description
delColNum	int	No	The first column to delete. Columns are numbered from left to right, starting with 0.
numDelCols	int	No	The number of columns to delete.

## DeleteRows

### Description

Deletes rows from a table. Like the Delete command in the FrameMaker product user interface, DeleteRows() does not allow you to delete more than one type of row at time.

The range of rows you specify must be all body rows, all header rows, or all footer rows.

The method deletes `refRow` and `(numDelRows - 1)` rows below it.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the requested operation.
Constants.FE_BadOperation (-27)	The method specified an invalid operation.
Constants.FE_BadDocId (-2)	invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_OutOfRange (-7)	The <code>refRow</code> parameter does not specify a row in the table, or the specified range includes more than one type of row (for example, header rows and body rows).

### Returns

int

### Syntax

```
DeleteRows (refRow, numDelRows)
```



## Parameters

Parameter name	Data Type	Optional	Description
refRow	<i>Row</i>	No	The first row to delete.
numDelRows	int	No	The number of rows to delete, including the value passed in refRow.

## GetTableColumnByColNum

### Description

Get column object of a table based on the column number.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadOperation (-27)	The method specified an invalid operation.
Constants.FE_OutOfRange (-7)	The specified row or column is invalid.
Constants.FE_BadParameter (-43)	The parameter specified is invalid.

### Returns

*Column*

### Syntax

`GetTableColumnByColNum (docId, tableId, colNum)`

## Parameters

Parameter name	Data Type	Optional	Description
docId	int	No	The id of the document containing the table
tableId	int	No	The id of the table for the required column object
colNum	int	No	Index of the required column

## MakeTblSelection

### Description

Selects a range of cells in a table.

To select an entire table, including the table title, set the `topRow` parameter to `Constants.FF_SELECT_WHOLE_TABLE (0x80000000)`. `MakeTblSelection()` ignores the values for the other parameters.

**NOTE:** `MakeTblSelection()` cannot select different types of rows at the same time, unless you set `topRow` to `Constants.FF_SELECT_WHOLE_TABLE (0x80000000)` or you set `topRow` and `bottomRow` to select one or more entire columns.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	invalid document.
<code>Constants.FE_BadObjId (-3)</code>	Invalid object.
<code>Constants.FE_BadOperation (-27)</code>	The method specified an invalid operation.
<code>Constants.FE_OutOfRange (-7)</code>	The specified row or column is invalid.
<code>Constants.FE_BadParameter (-43)</code>	The parameter specified is invalid.

### Returns

int

## Syntax

```
MakeTblSelection(topRow, bottomRow, leftCol, rightCol)
```

## Parameters

Parameter name	Data Type	Optional	Description
topRow	int	No	The number of the first row in the selection. The rows are numbered from top to bottom, starting with 0 (including heading rows). To select the entire table, specify <code>Constants.FF_SELECT_WHOLE_TABLE (0x80000000)</code> .
bottomRow	int	No	The number of the last row in the selection.
leftCol	int	No	The number of the leftmost column in the selection. The columns are numbered from left to right, starting with 0.
rightCol	int	No	The number of the rightmost column in the selection.

## Delete

### Description

Deletes the specified table.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified table.

See [GetProps](#) under the AFrame class for more information.

**Returns***PropVals***Syntax**`GetProps()`**SetProps****Description**

The `SetProps()` method sets the properties of the specified table.

See [SetProps](#) under the `AFrame` class for more information.

**Returns**

Void

**Syntax**`SetProps(setVal)`**Parameters**

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

**ObjectValid****Description**

Returns 1 if the table is valid, else returns 0.

See [ObjectValid](#) under the `AFrame` class for more information.

**Returns**

int

**Syntax**`ObjectValid()`

## TblFmt

The method uses an `FO_TblFmt` object to represent each table format in a document.

### Delete

Deletes the specified table format.

See [Delete](#) under the `AFrame` class.

### GetProps

See [GetProps](#) under the `AFrame` class.

### SetProps

See [SetProps](#) under the `AFrame` class.

### ObjectValid

See [ObjectValid](#) under the `AFrame` class.

## TextFrame

### Delete

#### Description

Deletes the specified text frame.

See [Delete](#) under the `AFrame` class for more information.

#### Returns

`int`

#### Syntax

```
Delete()
```

### GetText

#### Description

Gets the text from the text frame.

See [GetText](#) under the Cell class for more information.

### Returns

*TextItems*

### Syntax

GetText (flags)

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified text frame.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

GetProps ()

## SetProps

### Description

The `SetProps()` method sets the properties of the specified text frame.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

SetProps (setVal)

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the text frame is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## TextItems

An Array of [TextItem](#) objects with integer indexing and a *length* property.

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

[TextItems](#)

#### Syntax

```
concat (value)
```

## Parameters

Parameter name	Data Type	Optional	Description
value	<i>TextItems</i>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

*TextItem*

### Syntax

pop ()

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

push (value)

### Parameters

Parameter name	Data Type	Optional	Description
value	<i>TextItem</i>	No	The value to push into the array.



# TextLine

## Delete

### Description

Deletes the specified text line.

See [Delete](#) under the AFrame class for more information.

### Returns

`int`

### Syntax

```
Delete()
```

## GetText

### Description

Gets the text from the text line.

See [GetText](#) under the Cell class for more information.

### Returns

[TextItems](#)

### Syntax

```
GetText(flags)
```

### Parameters

Parameter name	Data Type	Optional	Description
flags	<code>int</code>	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified text line.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

### Syntax

```
GetProps ()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the specified text line.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the text line is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

# TextLoc

## TextLoc

### Description

TextLoc specifies a location within the text of a paragraph or a graphic text line.

### Returns

*TextLoc*

### Syntax

TextLoc(obj, offset)

### Parameters

Parameter name	Data Type	Optional	Description
obj	any	No	FO_Pgf or FO_TextLine object.
offset	int	No	Characters from the beginning of the object.

# TextRange

## TextRange

### Description

TextRange specifies a text range. A text range can span paragraphs. However, it cannot span graphic text lines or flows.

**NOTE:** *beg.offset* and *end.offset* fields of a *TextRange* object can specify offsets relative to the beginning and end of an object. In addition, they can use the special value *Constants.FV\_OBJ\_END\_OFFSET* (0x50000000). **NOTE:** *Constants.FV\_OBJ\_END\_OFFSET* (0x50000000) specifies the offset of the last character in the object containing the text range.

### Returns

*TextRange*

### Syntax

TextRange(beg, end)

## Parameters

Parameter name	Data Type	Optional	Description
beg	<i>TextLoc</i>	No	The beginning of the range.
end	<i>TextLoc</i>	No	The end of the range.

## TiApiClient

An `FO_TiApiClient` object represents text imported by an FDK client.

### GetText

See [GetText](#) under the Cell class.

### GetProps

See [GetProps](#) under the AFrame class.

### SetProps

See [SetProps](#) under the AFrame class.

### ObjectValid

See [ObjectValid](#) under the AFrame class.

## DeleteTextInsetContents

### Description

This method deletes the text in a text inset. You must unlock a text inset before you call this function to delete its contents. After you are done, you must relock the text inset.

Call this method on the text inset for which you want to delete the contents.

If this method fails, the method assigns one of the following values to `FA_errno`.

Error	Meaning
Constants.FE_BadDelete (-17)	Specified text couldn't be deleted

Error	Meaning
Constants.FE_BadDocId (-2)	Invalid document ID
Constants.FE_BadObjId (-3)	Invalid object ID
Constants.FE_BadSelectionForOperation (-59)	The specified text inset is locked

### Returns

int

### Syntax

DeleteTextInsetContents ()

### UpdateTextInset

#### Description

Updates the contents of a stale text inset. It determines whether an inset is stale by comparing the inset's `LastUpdate` property with the modification date of the inset's source file. `UpdateTextInset ()` does not update a text inset unless it is stale. To make a text inset stale, set its `LastUpdate` property to 0. `UpdateTextInset ()` does not update graphic insets (`FO_Inset` objects).

Call this method on the text inset to be updated.

This method returns `FE_Success` if it succeeds or `FE_SomeUnresolved` if some text insets were unresolved. If `UpdateTextInset ()` fails, it assigns one of the following values to `FA_errno`.

Error	Meaning
Constants.FE_BadDocId (-2)	Invalid document ID
Constants.FE_BadFileType (-83)	The inset specifies a file that does not match the import type (for example, the inset imports a binary document but the file is a text file or doesn't exist)
Constants.FE_SomeUnresolved (-22)	Some text insets were unresolved
Constants.FE_WrongProduct (-60)	Product doesn't support the specified operation
Constants.FE_SystemError (-42)	Couldn't allocate memory

**Returns**

int

**Syntax**

UpdateTextInset ()

**ConvertToText****Description**

This method is called upon a text inset (Examples: `TiApiClient()`, `TiFlow()`, `TiText()`, `TiTextTable()`) object and converts it to a text. This method deletes the text inset and removes the locked text range around it without modifying the content.

**Returns**

Void

**Syntax**

ConvertToText ()

**Delete**

See [Delete](#) under the AFrame class.

**DeletePropByName****Description**

Deletes an inset facet. Except for inset facets, all properties are identified by integer constants. Facets are the only properties identified by name.

If `DeletePropByName()` fails, the method assigns one of the following values to `FA_errno`:

Error	Meaning
<code>Constants.FE_BadPropNum (-4)</code>	Specified property number is invalid
<code>Constants.FE_BadDocId (-2)</code>	Invalid document ID
<code>Constants.FE_BadObjId (-3)</code>	Invalid object ID

Error	Meaning
<code>Constants.FE_BadPropType (-5)</code>	Incorrect property type for this function

**Returns**

Void

**Syntax**`DeletePropByName (propName)`**Parameters**

Parameter name	Data Type	Optional	Description
<code>propName</code>	<code>string</code>	No	The name of the property to delete

**TiFlow**

An `FO_TiFlow` object represents text imported from a FrameMaker document or a MIF file.

**GetText**

See [GetText](#) under the Cell class.

**GetProps**

See [GetProps](#) under the AFrame class.

**SetProps**

See [SetProps](#) under the AFrame class.

**ObjectValid**

See [ObjectValid](#) under the AFrame class.

**DeleteTextInsetContents**

See [DeleteTextInsetContents](#) under the TiApiClient class.

## UpdateTextInset

See [UpdateTextInset](#) under the TiApiClient class.

## ConvertToText

### Description

This method is called upon a text inset (Examples: TiApiClient, TiFlow, TiText, TiTextTable) object and converts it to a text. This method deletes the text inset and removes the locked text range around it without modifying the content.

### Returns

Void

### Syntax

```
ConvertToText ()
```

## Delete

See [Delete](#) under the AFrame class.

## DeletePropByName

See [DeletePropByName](#) under the TiApiClient.

## TiText

An `FO_TiText` object represents text imported from a text file.

### GetText

See [GetText](#) under the Cell class.

### GetProps

See [GetProps](#) under the AFrame class.

### SetProps

See [SetProps](#) under the AFrame class.



## ObjectValid

See [ObjectValid](#) under the AFrame class.

## DeleteTextInsetContents

See [DeleteTextInsetContents](#) under the TiApiClient class.

## UpdateTextInset

See [UpdateTextInset](#) under the TiApiClient class.

## ConvertToText

### Description

This method is called upon a text inset (Examples: `TiApiClient`, `TiFlow`, `TiText`, `TiTextTable`) object and converts it to a text. This method deletes the text inset and removes the locked text range around it without modifying the content.

### Returns

Void

### Syntax

```
ConvertToText ()
```

## Delete

See [Delete](#) under the AFrame class.

## DeletePropByName

See [DeletePropByName](#) under the TiApiClient class.

## TiTextTable

An `FO_TiTextTable` object represents text imported from a text file into a table.

## GetText

See [GetText](#) under the Cell class.

## GetProps

See [GetProps](#) under the AFrame class.

## SetProps

See [SetProps](#) under the AFrame class.

## ObjectValid

See [ObjectValid](#) under the AFrame class.

## DeleteTextInsetContents

See [DeleteTextInsetContents](#) under the TiApiClient class.

## UpdateTextInset

See [UpdateTextInset](#) under the TiApiClient class.

## ConvertToText

### Description

This method is called upon a text inset (Examples: `TiApiClient`, `TiFlow`, `TiText`, `TiTextTable`) object and converts it to a text. This method deletes the text inset and removes the locked text range around it without modifying the content.

### Returns

Void

### Syntax

```
ConvertToText()
```

## Delete

See [Delete](#) under the AFrame class.

## DeletePropByName

See [DeletePropByName](#) under the TiApiClient.

## TypedVals

An Array of *TypedVal* objects with integer indexing and a *length* property.

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

*TypedVal*

#### Syntax

```
concat (value)
```

#### Parameters

Parameter name	Data Type	Optional	Description
value	<i>TypedVal</i>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

#### Returns

*TypedVal*

#### Syntax

```
pop ()
```

### push

#### Description

Returns the new length of the array.

**Returns**

number

**Syntax**`push(value)`**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>TypedValue</i>	No	The value to push into the array.

**UBytes**

An Array of objects with integer indexing and a *length* property.

**concat****Description**

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

**Returns***UBytes***Syntax**`concat(value)`**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>UBytes</i>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

`uint`

### Syntax

```
pop()
```

## push

### Description

Returns the new length of the array.

### Returns

`number`

### Syntax

```
push(value)
```

### Parameters

Parameter name	Data Type	Optional	Description
value	<code>uint</code>	No	The value to push into the array.

## Uints

An Array of objects with integer indexing and a *length* property.

## concat

### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

**Returns***UInts***Syntax**`concat (value)`**Parameters**

Parameter name	Data Type	Optional	Description
value	<i>UInts</i>	No	Can also be arrays.

**pop****Description**

Removes the last element from the array.

**Returns**`uint`**Syntax**`pop ()`**push****Description**

Returns the new length of the array.

**Returns**`number`**Syntax**`push (value)`

## Parameters

Parameter name	Data Type	Optional	Description
value	uint	No	The value to push into the array.

## UnanchoredFrame

### Delete

#### Description

Deletes the specified unanchored frame.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

### GetProps

#### Description

The `GetProps()` method retrieves the properties of the specified unanchored frame.

See [GetProps](#) under the AFrame class for more information.

#### Returns

*PropVals*

#### Syntax

```
GetProps()
```

### SetProps

#### Description

The `SetProps()` method sets the properties of the specified unanchored frame.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the unanchored frame is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Var

### Delete

#### Description

Deletes the specified var object.

See [Delete](#) under the AFrame class for more information.

### Returns

int



## Syntax

Delete()

## GetText

### Description

Gets the text from the var object.

See [GetText](#) under the Cell class for more information.

### Returns

*TextItems*

## Syntax

GetText(flags)

## Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified var object.

See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

## Syntax

GetProps()

## SetProps

### Description

The `SetProps()` method sets the properties of the specified var object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the var object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## VarFmt

### Delete

#### Description

Deletes the specified varfmt object.

See [Delete](#) under the AFrame class for more information.

### Returns

int

## Syntax

```
Delete()
```

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified varfmt object. See [GetProps](#) under the AFrame class for more information.

### Returns

*PropVals*

## Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the specified var object. See [SetProps](#) under the AFrame class for more information.

### Returns

Void

## Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<i>PropVals</i>	No	The property list.

## ObjectValid

### Description

Returns 1 if the varfmt object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

```
ObjectValid()
```

## XRef

**Delete****Description**

Deletes the specified XRef object.

See [Delete](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

```
Delete()
```

**GetText****Description**

Gets the text from the XRef object.

See [GetText](#) under the Cell class for more information.

**Returns**

*TextItems*

**Syntax**

```
GetText(flags)
```

## Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified XRef object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the specified XRef object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the xref object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## XRefFmt

### Delete

#### Description

Deletes the specified XRefFmt object.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

### GetProps

#### Description

The `GetProps()` method retrieves the properties of the specified XRefFmt object.

See [GetProps](#) under the AFrame class for more information.

#### Returns

*PropVals*

#### Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the specified `XRefFmt` object.

See [SetProps](#) under the `AFrame` class for more information.

### Returns

`Void`

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>setVal</code>	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the `XRefFmt` object is valid, else returns 0.

See [ObjectValid](#) under the `AFrame` class for more information.

### Returns

`int`

### Syntax

```
ObjectValid()
```

## app

### Close

See [Close](#) under the `Book` class.

## GetNamedObject

### Description

Gets the object with a specified name (`Name` property) and object type.

If an error occurs, this method returns the the object, or 0. If this method fails, the method assigns one of the following values to `FA_errno`:

Error	Meaning
<code>Constants.FE_BadDocId</code> (-2)	Invalid document.
<code>Constants.FE_TypeUnName</code> <code>d</code> (-19)	Objects of the specified type do not have names.
<code>Constants.FE_NameNotFound</code> (-20)	Object with the specified name and type does not exist in the specified document.

### Returns

Object

### Syntax

`GetNamedObject(objType, name)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>objType</code>	<code>int</code>	No	Type of the object to get.
<code>name</code>	<code>string</code>	No	Name of the object to get.

## GetNamedBook

### Description

Gets the book object with a specified name.

If this method fails, the method assigns the following value to `FA_errno`.



Error	Meaning
Constants.FE_NameNotFound (-20)	The book with the specified name does not exist.

**Returns***Book***Syntax**

GetNamedBook (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the book object you want to get.

**GetNamedCommand****Description**Gets a `FO_command` object with the specified command string.If this method fails, the method assigns the following value to `FA_errno`.

Error	Meaning
Constants.FE_NameNotFound (-20)	The object with the specified name does not exist.

**Returns***Command***Syntax**

GetNamedCommand (name)

## Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the command object you want to get.

## GetNamedMenu

### Description

Gets a menu object with the specified menu string.

If this method fails, the method assigns the following value to `FA_errno`.

Error	Meaning
<code>Constants.FE_NameNotFound</code> (-20)	A menu with the specified name does not exist.

### Returns

*Menu*

### Syntax

`GetNamedMenu (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the menu object you want to get.

## GetNamedMenuItemSeparator

### Description

Gets a menu item separator object with the specified menu item separator string.

If this method fails, the method assigns the following value to `FA_errno`.

Error	Meaning
Constants.FE_NameNotFound (-20)	A menu item separator with the specified name does not exist.

**Returns***MenuItemSeparator***Syntax**

GetNamedMenuItemSeparator (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name of the menu item separator you want to get.

**NewNamedObject****Description**

Creates a new objects and returns it.

If this method fails, the method assigns one of the following values to `FA_errno`.

Error	Meaning
FE_BadName (-33)	Specified name for the new object is invalid.
FE_BadNew (-23)	Object cannot be created.
FE_DupName (-32)	Specified name for the new object belongs to an existing object.

**Returns**

Object

**Syntax**

NewNamedObject (objType, name)

## Parameters

Parameter name	Data Type	Optional	Description
objType	int	No	The type of the object you want to create.
name	string	No	The name of the object you want to create.

## NewNamedBook

### Description

Creates a new book objects and returns it.

If this method fails, the method assigns one of the following values to `FA_errno`.

Error	Meaning
FE_BadName (-33)	Specified name is invalid.
FE_BadNew (-23)	Object cannot be created.
FE_DupName (-32)	Specified name for the new object belongs to an existing object.

### Returns

*Book*

### Syntax

`NewNamedBook (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the book object to create.

## NewNamedCommand

### Description

Creates a new command object and returns it

If this method fails, the method assigns one of the following values to `FA_errno`.

Error	Meaning
FE_BadName (-33)	Specified name is invalid.
FE_BadNew (-23)	Command cannot be created.
FE_DupName (-32)	Specified name for the new command belongs to an existing command.

**Returns***Command***Syntax**

NewNamedCommand (name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	Name of the new command object to create.

**NewNamedMenu****Description**

Creates a menu object with the specified name and returns it

If this method fails, the method assigns one of the following values to `FA_errno`.

Error	Meaning
FE_BadName (-33)	Specified name is invalid.
FE_BadNew (-23)	Menu cannot be created.
FE_DupName (-32)	Specified name for the new menu belongs to an existing menu.

**Returns***Menu***Syntax**

NewNamedMenu (name)

## Parameters

Parameter name	Data Type	Optional	Description
name	string	No	Name of the menu object to create.

## NamedMenuItemSeparator

### Description

Creates a new menu item separator with the specified name and returns it

If this method fails, the method assigns one of the following values to `FA_errno`.

Error	Meaning
FE_BadName (-33)	Specified name is invalid.
FE_BadNew (-23)	Menu item separator cannot be created.
FE_DupName (-32)	Specified name for the new menu item separator belongs to an existing menu item separator.

### Returns

*MenuItemSeparator*

### Syntax

`NamedMenuItemSeparator(name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	Name of the new menu item separator object.

### GetProps

See [GetProps](#) under the `AFrame` class.

## SetProps

See [SetProps](#) under the AFrame class.

## ObjectValid

See [ObjectValid](#) under the AFrame class.

# global

## Alert

### Description

Displays an alert box with a message. Depending on the constant you specify for the `type` parameter, the alert displays either the OK and Cancel buttons, Yes and No buttons, or a Continue button.

The `type` parameter takes one of the following values:

Constant	Value
Constants.FF_ALERT_OK_DEFAULT (0)	Displays the OK and Cancel buttons; OK is the default.
Constants.FF_ALERT_CANCEL_DEFAULT (1)	Displays the OK and Cancel buttons; Cancel is the default.
Constants.FF_ALERT_CONTINUE_NOTE (2)	Displays the OK button.
Constants.FF_ALERT_CONTINUE_WARN (3)	Displays the OK button with a warning indicator.
Constants.FF_ALERT_YES_DEFAULT (4)	Displays the Yes and No buttons; Yes is the default.
Constants.FF_ALERT_NO_DEFAULT (5)	Displays the Yes and No buttons; No is the default.

The method returns 0 if the user clicked OK, Continue, or Yes.

The method returns -1 if the user clicked Cancel or No.

### Returns

`int`

## Syntax

```
Alert(message, type)
```

## Parameters

Parameter name	Data Type	Optional	Description
message	string	No	The message that appears in the dialog box. Messages longer than 255 characters, are truncated.
type	int	No	The dialog box type.

## Alive

### Description

Checks whether the current asynchronous client has a connection with a FrameMaker process. Call this method after registering the asynchronous client using `WinConnectSession()`.

The method returns a positive integer if there is a current connection to a FrameMaker process. Otherwise it returns 0.

### Returns

```
int
```

### Syntax

```
Alive()
```

## CallClient

### Description

Allows a client to call another client. The method is useful for calling FrameMaker clients, such as the structure generator and the element catalog manager.

On success, the method returns `FE_Success` or the value specified by the target client's last call to `ReturnValue()`.

**NOTE:** Calls to the structure generator always return `FE_Success` no matter what string is passed to it as an argument.

On failure, the method assigns the following value to `FA_errno`:



Error	Reason
Constants.FE_NameNotFound (-20)	There is no client with the specified name in the current FrameMaker productsession.
Constants.FE_BadParameter (-43)	For the TableSort client only: One of the arguments is invalid. For example, you gave a value for the sort key that is greater than the number of columns or rows in the current table selection, or you have no table cells selected.

**Returns**

int

**Syntax**

```
CallClient(cname, arg)
```

**Parameters**

Parameter name	Data Type	Optional	Description
cname	string	No	The registered name of the target client.
arg	string	No	A string that is passed to the target client.

**Example: XLIFF Export**

The following example illustrates how you can use the `CallClient` method to export your documents into XLIFF format for translation.

**Syntax**

```
CallClient("XLIFF-Client,  
"action=xliffexport;src=SOURCEFILENAME;xts=XTSFILENAME;lang=SOURCELANG;out=OUTFILENAME");
```

**Parameters**

Parameter name	Description
action	A fixed value of <code>xliffexport</code> .
src	Fully qualified path of root map or file to convert. Supported file type extension: <code>.dita</code> , <code>.ditamap</code> , <code>.xml</code> , <code>.book</code> , <code>.mif</code> .

Parameter name	Description
xts	Fully qualified path to XTS file (XTS - XLIFF Translation Settings). <b>NOTE:</b> XTS settings for MIF can be configured through the <i>XLIFF Export</i> dialog.
lang	Source language code in ISO format, such as en-US.
out	Fully qualified output file name, which is a .zip file.

**Returns**

0

Success

-1 to -9

Process cancelled due to missing or wrong parameters.

-10

Errors occurred during the conversion process.

**Sample code:**

```

var XLIFFCLIENTNAME="XLIFF-Client";
ExportXliff() ;
ExportXliff = function()
{
    var inFile = new File().openDlg ("select input file", "All files:*.*",
false);
    if (inFile == null)
        return ;
    var outFile = new File(inFile.fsName + ".zip").saveDlg("select output
file", "ZIP Files:.zip");
    if (outFile == null)
        return ;
    var xtsFile = new File(app.HomeDir +
"\Translation\XLIFF\config\default.xts").openDlg("select XTS file",
"XTS file:.xts", false);
    if (xtsFile == null)
        return ;
    var sourceLang = "en-US";
    var cmd = "action=xliffexport;";
    cmd += "src=" + inFile.fsName + ";";
    cmd += "xts=" + xtsFile.fsName + ";";
    cmd += "lang=" + sourceLang + ";";
    cmd += "out=" + outFile.fsName + ";";
    var result = CallClient (XLIFFCLIENTNAME, cmd);
    if (result == 0)
    {

```

```

        $.writeln("Success");
    }
    else if (result < 0 && result > -10)
    {
        $.writeln("parameter errors. process cancelled");
    }
    else if (result <= -10)
    {
        $.writeln("conversion errors occurs. Check log file");
    }
}

```

### Example: XLIFF Import

The following example illustrates how you can use the `CallClient` method to import converted XLIFF files back into your source document.

#### Syntax

```

CallClient("XLIFF-Client,
"action=xliffimport;src=XLIFFZIPPATH;out=OUTDIRECTORY");

```

#### Parameters

Parameter name	Description
action	A fixed value of <code>xliffimport</code> .
src	Fully qualified path of the root zip file containing the translated XLIFF files.
out	Fully qualified path to output directory. <b>CAUTION:</b> Ensure that the output directory is empty, else existing files are overwritten.

#### Returns

0

Success

-1 to -9

Process cancelled due to missing or wrong parameters.

-10

Errors occurred during conversion process.

#### Sample code:

```

var XLIFFCLIENTNAME="XLIFF-Client";
ImportXliff();
ImportXliff = function()

```

```

{
    //Select source file (.book; .fm; .dita; .ditamap; .xml)
    var inFile = new File().openDlg ("select XLIFF ZIP file", "ZIP
Files:*.zip", false);
    if (inFile == null)
        return ;
    //!!!!!!!!!!!! CAUTION !!!!!!!!!!!!!!!
    //!!!!!!select empty directory, otherwise FILES may be overwritten!!!!
    var ouDirectory = new Folder(inFile.parent).selectDlg("select output
directory");
    if (ouDirectory == null)
        return ;
    var cmd = "action=xliffimport;";
    cmd += "src=" + inFile.fsName+ ";";
    cmd += "out=" + ouDirectory.fsName + ";";
    var result = CallClient (XLIFFCLIENTNAME, cmd);
    if (result == 0)
    {
        $.writeln("Success");
    }
    else if (result < 0 && result > -10)
    {
        $.writeln("parameter errors. process cancelled");
    }
    else if (result <= -10)
    {
        $.writeln("conversion errors occurs. Check log file");
    }
}
}

```

## CallClientEx

### Description

Allows a client to call another client. The method is useful for calling FrameMaker clients, such as the structure generator and the element catalog manager.

On success, the method returns `FE_Success` or the value specified by the target client's last call to `ReturnValue()`.

**NOTE:** Calls to the structure generator always return `FE_Success` no matter what string is passed to it as an argument.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_NameNotFoun d (-20)	There is no client with the specified name in the current FrameMaker productsession.

Error	Reason
Constants.FE_BadParameter (-43)	For the TableSort client only: One of the arguments is invalid. For example, you gave a value for the sort key that is greater than the number of columns or rows in the current table selection, or you have no table cells selected.

**Returns***TypedVal***Syntax**

```
CallClientEx(cname, arg)
```

**Parameters**

Parameter name	Data Type	Optional	Description
cname	string	No	The registered name of the target client.
arg	string	No	A string that is passed to the target client.

**CheckStatus****Description**

Checks the scripts returned by `Open()`, `Import()`, `Save()`, and `UpdateBook()` to determine if a specified status bit is set.

Returns 1 if the bit is set, else returns 0.

**Returns**

```
int
```

**Syntax**

```
CheckStatus(p, statusBit)
```

## Parameters

Parameter name	Data Type	Optional	Description
p	<i>PropVals</i>	No	The property list returned by Open(), Save(), Import(), or UpdateBook().
statusBit	int	No	The status bit to test.

## ClientDir

### Description

Returns the name of the current ESTK client's directory.

*NOTE: Use Free() to free the string returned by ClientDir() when you are done with it.*

The method assigns the following value to `FA_errno`, on failure:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

### Returns

string

### Syntax

```
ClientDir()
```

## ClientName

### Description

Returns the registered name of the current client (the client that calls `ClientName()`).

*NOTE: Use Free() to free the string returned by ClientDir() when you are done with it.*

The method assigns the following value to `FA_errno`, on failure:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

**Returns**

string

**Syntax**

```
ClientName ( _ )
```

**CMSRegister****Description**

Registers a CMS client.

**Returns**

CMS Object

**Syntax**

```
CMSRegister (cmsName);
```

**Parameters**

Parameter name	Data Type	Optional	Description
cmsName	int	No	Name of the CMS to register

If CMSRegister() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSNameAlreadyRegistered	The method attempts to register a CMS that is already registered.
Constants.FE_BadParameter	The function call specified an invalid parameter.

**CMSGetCmsIdFromName****Description**

Gets the CMS registration id from CMS name.

**Returns**

CMS Object

**Syntax**

```
CMSGetCmsIdFromName (cmsName);
```

**Parameters**

Parameter name	Data Type	Optional	Description
cmsName	int	No	The Name of the CMS

If CMSGetCmsIdFromName() fails, the method assigns following values to FA\_erno:

Error	Reason
Constants.FE_BadParameter	The function call specified an invalid parameter.

**CMSAddMenuEntry****Description**

Adds a custom menu entry in the context menu within the FrameMaker interface.

**Returns**

CMS Object

**Syntax**

```
CMSAddMenuEntry (menuId,menuEntry);
```

**Parameters**

Parameter name	Data Type	Optional	Description
menuId	int	No	The ID of the Parent menu
menuEntry	int	No	The F_CMSMenuItemT structure describes a custom menu definition

If CMSAddMenuEntry() fails, the method assigns following values to FA\_erno:



Error	Reason
Constants.FE_CMSBadObjectID	The client specified an invalid menu ID.
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CustomDoc

### Description

Creates a new custom document using the FrameMaker product's default new document template.

The sidedness parameter can have one of the following values:

Vaule	Page Characteristics
Constants.FF_Custom_SingleSided (0)	Single sided.
Constants.FF_Custom_FirstPageRight (1)	Double-sided, starting with a right page.
Constants.FF_Custom_FirstPageLeft (2)	Double-sided, starting with a left page.

The method returns the new document on success.

The method assigns the following value to `FA_errno`, on failure:

Error	Reason
Constants.FE_WrongProduct (-60)	The product version does not support this operation.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.

### Returns

*Doc*

### Syntax

```
CustomDoc(width, height, numCols, columnGap, topMargin, botMargin,
leftinsideMargin, rightoutsideMargin, sidedness, makeVisible)
```

## Parameters

Parameter name	Data Type	Optional	Description
width	int	No	The document page width.
height	int	No	The document page height.
numCols	int	No	The default number of columns.
columnGap	int	No	The default column spacing.
topMargin	int	No	The document page top margin.
botMargin	int	No	The document page bottom margin.
leftinsideMargin	int	No	The left margin for single-sided documents, or the inside margin for double-sided documents.
rightoutsideMargin	int	No	The right margin for single-sided documents, or the outside margin for double-sided documents.
sidedness	int	No	A constant that specifies whether the document is single-sided or doublesided and on which side the document starts.
makeVisible	bool	No	Specifies whether the document is visible after it is created. Set to <code>True</code> to make the document visible.

## DefineMenu

### Description

Defines a menu (FO\_Menu object). After you define a menu, you can add it to a menu or a menu bar with `AddMenuToMenu()`.

If you call `DefineMenu()` and specify the name of a menu that is already defined in the user's menu configuration files, the FrameMaker product gives precedence to the definition in the configuration files. If the configuration files assign a label to the menu, the FrameMaker product uses it instead of the one you specify. If the user has already defined a menu with the name specified by `name`, `DefineMenu()` ignores the label parameter and uses the label specified by the user.

The method returns the new menu on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running
Constants.FE_BadParameter (-43)	Parameter has an invalid value.
Constants.FE_BadOperation (-27)	Parameters specify an invalid operation.
Constants.FE_NotMenu (-72)	<code>name</code> specifies a command; cannot redefine a command as a menu.
Constants.FE_SystemError (-42)	System error.

## Returns

*Menu*

## Syntax

```
DefineMenu(name, label)
```

## Parameters

Parameter name	Data Type	Optional	Description
name	string	No	A unique name for the menu. If the user or an ESTK client has already defined a command or menu with this name, the new menu replaces it.
label	string	No	The title of the menu as it appears on the menu bar or menu.

## DisconnectFromSession

### Description

Ends communication with a FrameMaker product process.

The method returns `FE_Success` if it succeeds, or a system error code if an error occurs.

**Returns**`int`**Syntax**`DisconnectFromSession()`**Err****Description**

Prints the client's name and a message to the console.

**Returns**`Void`**Syntax**`Err(message)`**Parameters**

Parameter name	Data Type	Optional	Description
message	string	No	The message to print.

**FamilyFonts****Description**

Returns a `Fonts` object with the permutations of angles, variations, and weights available for a specified font family.

**Returns**`Fonts`**Syntax**`FamilyFonts(family)`

## Parameters

Parameter name	Data Type	Optional	Description
family	int	No	The index of the font family (in the list of fonts in the session).

## GetKeyCatalog

### Description

Finds a key catalog with the specified 'tag'.

### Returns

KeyCatalog

### Syntax

```
GetKeyCatalog(tag);
```

### Parameters

Parameter name	Data Type	Optional	Description
tag	string	No	The tag of the new Key Catalog being created.

If GetKeyCatalog() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadName	The tag provided is not valid or the key catalog with this tag does not exist.

## GetEncodingForFamily

### Description

Returns the encoding that FrameMaker uses for the font family.

Returns one of the following strings indicating the encoding for the font:

Value	Meaning
FrameRoman	Roman Text.
JISX0208.ShiftJIS	Japanese Text.
BIG5	Traditional Chinese Text.
GB2312-80.EUC	Simplified Chinese Text.
KSC5601-1992	Korean Text.
Multiple	More than one encoding for the font family.

If the returned string is `Multiple`, the font family includes variations that are represented by different encodings. You should use `ApiFamilyFonts()` to get a list of the variations for the family. Then you can use `GetEncodingForFont()` to get the encoding for a specific variation.

### Returns

`string`

### Syntax

`GetEncodingForFamily(family)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>family</code>	<code>int</code>	No	The font family for which the encoding is to be retrieved.

## GetEncodingForFont

### Description

Returns the encoding that FrameMaker uses for a specific font with a specific combination of weight, angle, and variation.

Returns one of the following strings indicating the encoding for the font:

Value	Meaning
FrameRoman	Roman Text.

Value	Meaning
JISX0208.ShiftJIS	Japanese Text.
BIG5	Traditional Chinese Text.
GB2312-80.EUC	Simplified Chinese Text.
KSC5601-1992	Korean Text.

**Returns**

string

**Syntax**

GetEncodingForFont (font)

**Parameters**

Parameter name	Data Type	Optional	Description
font	<i>Font</i>	No	

**GetImportDefaultParams****Description**

Gets a default property list that you can use to call `Import()`.

The method returns a `PropVals` structure containing the properties retrieved.

For more information see [Import](#).

**Returns**

*PropVals*

**Syntax**

GetImportDefaultParams ()

**GetOpenDefaultParams****Description**

Gets a default property list that you can use to call `Open()`.

## Returns

*PropVals*

## Syntax

```
GetOpenDefaultParams()
```

## GetPropIndex

### Description

Gets the index of a property-value pair (PropVal structure) within a property list. `GetPropIndex()` is a convenience routine that makes it easier to manipulate the properties in a property list.

The method returns the index (in the property list) of the property's PropVal structure, or `Constants.FE_BadPropNum` (-4) if an error occurs.

### Returns

int

### Syntax

```
GetPropIndex(propVals, propNum)
```

### Parameters

Parameter name	Data Type	Optional	Description
propVals	<i>PropVals</i>	No	The property list.
propNum	int	No	The property for which the index is to be retrieved.

## GetSaveDefaultParams

### Description

Gets a default property list that you can use to call `Save()`.

On success, the method returns a `PropVals` list as shown in the following table.

The first value listed next to each property is the value that `GetSaveDefaultParams()` assigns to the property. The other values are values that you can assign to the property as desired.



Property	Values
AlertUserAboutFailure	<p>Specifies whether to notify user if something unusual happens during the Save operation. The value is one of:</p> <ul style="list-style-type: none"> <li>False: Do not notify user.</li> </ul> <p><i>True: Notify user.</i></p>
AutoBackupOnSave	<p>Specifies whether to create a backup file. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_SaveUserPrefAutoBackup(2) - Follow preference specified by the session's <i>AutoBackup</i> property.</li> </ul> <p><i>Constants.FV_SaveYesAutoBackup(0) - Make a backup.</i></p> <p><i>Constants.FV_SaveNoAutoBackup(1) - Do not make a backup.</i></p>
DontNotifyAPIClients	<p>Specifies whether to save the file without notifying other clients. The value is one of:</p> <ul style="list-style-type: none"> <li>False - Notify other clients.</li> </ul> <p><i>True - Do not notify other clients.</i></p>

Property	Values
FileType	<p>Specifies the type of file to save to. This file type must be one that FrameMaker saves natively. Note that HTML and XML are saved via filters, and so you must specify a filter hint string via <code>Constants.FS_SaveFileTypeHint(16)</code>. The value is one of:</p> <ul style="list-style-type: none"> <li> <p><code>Constants.FV_SaveFmtBinary(0)</code> - Save in Frame binary format for this version of FrameMaker.</p> <p><i>Constants.FV_SaveFmtBinary60(11)</i> - Save in binary format for FrameMaker 6.0.</p> <p><i>Constants.FV_SaveFmtInterchange(1)</i> - Save as MIF.</p> <p><i>Constants.FV_SaveFmtSgml(7)</i> - Save in SGML format.</p> <p><i>Constants.FV_SaveFmtText(6)</i> - Save in Text Only format.</p> <p><i>Constants.FV_SaveFmtXml(10)</i> - Save in XML format.</p> <p><i>Constants.FV_SaveFmtPdf(9)</i> - Save as PostScript, and then invoke Acrobat Distiller to create a PDF version of the document. This is the same as choosing PDF from the Format popup menu in the Save As dialog box.</p> <p><i>Constants.FV_SaveFmtFilter(8)</i> - Filter on save, using <i>Constants.FS_SaveFileTypeHint(16)</i> to determine the filter.</p> </li> </ul>
FileIsInUse	<p>Another user or session is recorded in the file's lock file. The value is one of:</p> <ul style="list-style-type: none"> <li> <p><code>Constants.FV_DoCancel(0)</code> - Cancel the Save operation.</p> <p><i>Constants.FV_DoShowDialog(4)</i> - Display the File In Use dialog box.</p> <p><i>Constants.FV_ResetLockAndContinue(7)</i> - Attempt to reset the file lock and save the document.</p> </li> </ul>

Property	Values
LockCantBeReset	<p>The user clicked Save Anyway in the File In Use dialog box, or the value of <code>Constants.FS_FileInUse</code> is set to <code>Constants.FV_ResetLockAndContinue(7)</code>, but the lock file cannot be reset. This is usually due to permissions in the lock file.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_DoCancel(0)</code> - Cancel the Save operation.</li> </ul> <p><i>Constants.FV_DoShowDialog(4) - Display the Cannot Lock File dialog box.</i></p> <p><i>Constants.FV_DoOK(1) - Save the document anyway.</i></p>
ModDateChanged	<p>The file has changed since the last time it was opened or saved in the current session. Somebody else has probably modified the file.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_DoCancel(0)</code> - Cancel the Save operation.</li> </ul> <p><i>Constants.FV_DoShowDialog(4) - Display the File Has Changed alert box.</i></p> <p><i>Constants.FV_DoOK(1) - Save the document anyway.</i></p>
SaveFileNotWritable	<p>The file permissions will not allow the file to be saved.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_DoCancel(0)</code> - Cancel the Save operation.</li> </ul> <p><i>Constants.FV_DoShowDialog(4) - Display the Cannot Lock File alert box.</i></p>
SaveFileTypeHint	<p>If <code>Constants.FS_FileType</code> is <code>Constants.FV_SaveFmtFilter(8)</code>, this string enables the FrameMaker product to call the correct filter. For example, use <code>0001ADBEHTML</code> to save as HTML or <code>0001ADBEXML</code> to save as XML.</p>

Property	Values
MakePageCount	<p>Specifies how to round the page count. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_UseCurrentSetting(0) - Use default specified by the document property, PageRounding.</li> </ul> <p><i>Constants.FV_DontChangePageCount(1) - Leave pages as is.</i></p> <p><i>Constants.FV_MakePageCountEven(2) - With odd number of pages, add a page to end of document.</i></p> <p><i>Constants.FV_MakePageCountOdd(3) - With even number of pages, add a page to end of document.</i></p> <p><i>Constants.FV_DeleteEmptyPages(4) - Remove extra pages at end of document.</i></p>
RetainNameStripe	<p>Specifies whether to change the name in document title bar to the name to which the file is saved. The value is one of:</p> <ul style="list-style-type: none"> <li>• False -Change the name in the title bar to the name in which the file is saved.</li> </ul> <p><i>True - Do not change the name in the title bar.</i></p>
SaveAsModeName	<p>Specifies where to get filename if Constants.FS_SaveMode(4) is set to Constants.FV_ModeSaveAs(1). The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_SaveAsNameProvided(2) - Save under the filename specified in the <i>saveAsName</i> parameter of <i>Save()</i>.</li> </ul> <p><i>Constants.FV_SaveAsUseFileName(1) - Save as name shown on the document title bar.</i></p> <p><i>Constants.FV_SaveAsNameAskUser(0) - Prompt user for name.</i></p>
SaveMode	<p>Specifies whether to use Save or Save As mode. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_ModeSaveAs(1) - Use Save As mode.</li> </ul> <p><i>Constants.FV_ModeSave(0) - Use Save mode.</i></p>

Property	Values
SaveTextExtraBlankLineAtEOP	<p>Specifies whether to add an extra line at the end of each paragraph if the file is being saved as Text Only. The value is one of:</p> <ul style="list-style-type: none"> <li>False - Do not add an extra line.</li> <li><i>True - Add an extra line.</i></li> </ul>
SaveTextTblSetting	<p>Specifies how to deal with tables if the file is being saved as Text Only. The value is one of:</p> <ul style="list-style-type: none"> <li>Constants.FV_SaveTblUserPref(0) - Use setting last specified in Save as Text dialog box.</li> <li><i>Constants.FV_SaveTblRowsAsPgfs(1) - Save each table cell as a paragraph row-by-row.</i></li> <li><i>Constants.FV_SaveTblColsAsPgfs(2) - Save each table cell as a paragraph column-by-column.</i></li> <li><i>Constants.FV_SaveSkipTbIs(3) - Omit tables from a Text Only file.</i></li> <li><i>Constants.FV_SaveTextTblCellSeparator - The character to write as a cell separator in the text file.</i></li> <li><i>Constants.FV_SaveTextTblRowColumnSeparator - The character to write as a row or column separator in the text file.</i></li> </ul>
SgmlSaveApplication	<p>Retained for compatibility. Use the StructuredSaveApplication property instead.</p>
StructuredSaveApplication	<p>The value is one of:</p> <ul style="list-style-type: none"> <li>String specifying the name of the structure application to use when saving a structured document as XML or SGML. This parameter takes precedence over any other structure application specification. If the specified application does not exist, the calling function will fail.</li> <li><i>Null - No application used.</i></li> </ul>
ShowSaveTextDialog	<p>Specifies whether to display dialog box if the file is being saved in Text Only format. The value is one of:</p> <ul style="list-style-type: none"> <li>False - Do not display dialog box.</li> <li><i>True - Display dialog box asking user whether to put paragraph returns at the end of each line.</i></li> </ul>

Property	Values
UpdateFRVList	<p>Specifies whether the file will be added to the list of files recently visited that appears in the File menu. This is set to False by default.</p> <p>The value is one of :</p> <ul style="list-style-type: none"> <li>False - Do not add the file to the list.</li> <li>True - Add the file to the list.</li> </ul>

On failure, the method sets the `len` field of the returned structure to 0.

### Returns

*PropVals*

### Syntax

`GetSaveDefaultParams()`

## GetSupportedEncodings

### Description

Returns the font encodings supported for the current session.

The following strings indicate the encoding for fonts:

Value	Meaning
FrameRoman	Roman Text.
JISX0208.ShiftJIS	Japanese Text.
BIG5	Traditional Chinese Text.
GB2312-80.EUC	Simplified Chinese Text.
KSC5601-1992	Korean Text.

### Returns

*Strings*

### Syntax

`GetSupportedEncodings()`

## GetUpdateBookDefaultParams

### Description

Gets a default property list that you can use to call UpdateBook().

### Returns

*PropVals*

### Syntax

```
GetUpdateBookDefaultParams ()
```

## GetNewXMLDefaultParams

### Description

Generates default open-parameters for NewXML().

### Returns

*PropVals*

### Syntax

```
GetNewXMLDefaultParams () ;
```

A property list (PropVals) with the properties shown in the following table.

Property	Instruction or situation and possible values
Constants.FS_Doctype	Specifies a doctype to be used for creating a new XML document
Constants.FS_PublicId	Specifies a public id to be used for creating a new XML document
Constants.FS_SystemId	Specifies a DTD-system id to be used for creating a new xml document
Constants.FS_Extension	Used to provide a custom extension for the new XML document (like Untitled1.dita). Without customization, FrameMaker determines the extension by itself based on file type.
Constants.FS_Visible	A boolean property that indicates if the new XML document shall be visible or hidden.

## IsEncodingSupported

### Description

Checks whether the specified encoding is supported for the current session. For example, unless FrameMaker is running on a system that supports Japanese text, Japanese encoding is not supported.

The `encodingName` parameter can be set to one of the following values:

Value	Meaning
FrameRoman	Roman Text.
JISX0208.ShiftJIS	Japanese Text.
BIG5	Traditional Chinese Text.
GB2312-80.EUC	Simplified Chinese Text.
KSC5601-1992	Korean Text.

Returns `True` if the specified encoding is supported for the current session, else returns `False`.

### Returns

`bool`

### Syntax

```
IsEncodingSupported(encodingName)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>encodingName</code>	<code>string</code>	No	The encoding that you want to test.

## LoadMenuCustomizationFile

### Description

Loads a menu customization file.

A menu customization file is a text file containing statements that change the menus and commands the user sees in FrameMaker. For example, a menu customization file can change the name of a command or move a command from one menu to another.

The method returns `FE_Success` on success.



On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_WrongProduct (-60)	The product version does not support this operation.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.
Constants.FE_BadOperation (-27)	Parameters specify an invalid operation.
Constants.FE_SystemError (-42)	System error.

### Returns

Void

### Syntax

```
LoadMenuCustomizationFile(pathname, silent)
```

### Parameters

Parameter name	Data Type	Optional	Description
pathname	string	No	The pathname of the menu customization file to load. If you specify only a filename, the function looks in the client directory. If <code>silent</code> is set to <code>False</code> , the pathname specified by <code>pathname</code> is used as the default in the Menu Customization File dialog box.
silent	bool	No	Specifies whether to display the Menu Customization File dialog box and allow the user to choose the file. To display the dialog box and allow the user to choose the file, specify <code>False</code> . To use the file specified by <code>pathname</code> without asking the user, specify <code>True</code> .

## NewKeyCatalog

### Description

Creates a new key catalog with the specified 'tag'.

### Returns

KeyCatalog

### Syntax

```
NewKeyCatalog (tag) ;
```

### Parameters

Parameter name	Data Type	Optional	Description
tag	string	No	The tag of the new Key Catalog being created.

If NewKeyCatalog() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadName	The tag provided is not valid.
Constants.FE_DupName	A key catalog for the tag provided already exists.

## NewXML

### Description

Creates a new, untitled XML.

### Returns

Doc

### Syntax

```
ApiNewXML (opennewXMLParams, newXMLReturnParams) ;
```

## Parameters

Parameter name	Data Type	Optional	Description
opennewXMLParamsp	<i>PropVals</i>	No	A property list telling the FrameMaker product how to open the file and how to respond to errors and other conditions. To use the default list, specify NULL.
newXMLOpenReturnParams	<i>PropVals</i>	No	A property list that returns the filename and provides information about how the FrameMaker product opened the file. It must be initialized before you call <code>F_ApiNewXMLOpen()</code> .

To get a property list to specify for the `newXMLParams` parameter, use `GetNewXMLDefaultParams`.

## Notification

### Description

Requests that the FrameMaker product notify your client whenever a specified event, or stage of an event, occurs.

**NOTE:** *If FrameMaker encounters an internal error and exits, it does not send any notification to your client about operations performed after the error occurred. For example, after an error, FrameMaker allows the user to save changes in open documents, but it does not notify any clients of the save operations.*

Many events have several notification points or stages for which you can request notification. The following table lists the notification points and the constants that specify them:

Event	Notification Point	Notification Constant
Frame binary document opened	<ul style="list-style-type: none"> <li>Before checking the type of file to be opened</li> <li><i>After checking the type of file to be opened</i></li> <li>Before opening the file</li> <li><i>After opening the file</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreFileType (9)</li> <li>Constants.FA_Note_PostFileType (10)</li> <li>Constants.FA_Note_PreOpenDoc (1)</li> <li>Constants.FA_Note_PostOpenDoc (2)</li> </ul>

Event	Notification Point	Notification Constant
MIF document opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li><i>After checking the type of file to be opened</i></li> <li><i>Before opening the file</i></li> <li><i>After opening the file</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li><i>Constants.FA_Note_PostFileType (10)</i></li> <li><i>Constants.FA_Note_PreOpenMIF (3)</i></li> <li><i>Constants.FA_Note_PostOpenMIF (4)</i></li> </ul>
SGML document opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li><i>After checking the type of file to be opened</i></li> <li><i>Before opening the file</i></li> <li><i>After opening the file</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li><i>Constants.FA_Note_PostFileType (10)</i></li> <li><i>Constants.FA_Note_PreOpenSGML (82)</i></li> <li><i>Constants.FA_Note_PostOpenSGML (83)</i></li> </ul>
XML document opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li><i>After checking the type of file to be opened</i></li> <li><i>Before opening the file</i></li> <li><i>After opening the file</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li><i>Constants.FA_Note_PostFileType (10)</i></li> <li><i>Constants.FA_Note_PreOpenXML (95)</i></li> <li><i>Constants.FA_Note_PostOpenXML (96)</i></li> </ul>
Filterable document opened	Before checking the type of the file to be opened	Constants.FA_Note_FilterIn (14)
Frame binary book opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li><i>After checking the type of file to be opened</i></li> <li><i>Before opening the file</i></li> <li><i>After opening the file</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li><i>Constants.FA_Note_PostFileType (10)</i></li> <li><i>Constants.FA_Note_PreOpenBook (16)</i></li> <li><i>Constants.FA_Note_PostOpenBook (17)</i></li> </ul>
MIF book opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li><i>After checking the type of file to be opened</i></li> <li><i>Before opening the file</i></li> <li><i>After opening the file</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li><i>Constants.FA_Note_PostFileType (10)</i></li> <li><i>Constants.FA_Note_PreOpenBookMIF (18)</i></li> <li><i>Constants.FA_Note_PostOpenBookMIF (19)</i></li> </ul>
User double-clicked to open a document in a book window	<ul style="list-style-type: none"> <li>• Before opening the file</li> <li><i>After opening the file</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreBookComponentOpen (76)</li> <li><i>Constants.FA_Note_PostBookComponentOpen (77)</i></li> </ul>

Event	Notification Point	Notification Constant
Generating a list or TOC for a document or a book	<ul style="list-style-type: none"> <li>Before generating the file <i>After generating the file</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreGenerate (78) <i>Constants.FA_Note_PostGenerate (79)</i></li> </ul>
Document saved in Frame binary format	<ul style="list-style-type: none"> <li>Before saving the document <i>After saving the document</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreSaveDoc (5) <i>Constants.FA_Note_PostSaveDoc (6)</i></li> </ul>
Document saved as MIF	<ul style="list-style-type: none"> <li>Before saving the file as MIF <i>After saving the file as MIF</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreSaveMIF (7) <i>Constants.FA_Note_PostSaveMIF (8)</i></li> </ul>
Document saved as SGML	<ul style="list-style-type: none"> <li>Before saving the file as SGML <i>After saving the file as SGML</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreSaveSGML (99) <i>Constants.FA_Note_PostSaveSGML (100)</i></li> </ul>
Document saved as XML	<ul style="list-style-type: none"> <li>Before saving the file as XML <i>After saving the file as XML</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreSaveXML (97) <i>Constants.FA_Note_PostSaveXML (98)</i></li> </ul>
Document saved as PDF	<ul style="list-style-type: none"> <li>Before specifying Acrobat settings and generating PostScript <i>After specifying Acrobat settings and generating PostScript</i></li> <li>Before distilling the post-script <i>After distilling the post-script</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreSaveAsPDFDialog (71) <i>Constants.FA_Note_PostSaveAsPDFDialog (72)</i></li> <li>Constants.FA_Note_PreDistill (73) <i>Constants.FA_Note_PostDistill (74)</i></li> </ul>
Document saved as filterable type	Before the document is saved	Constants.FA_Note_FilterOut (15)
Document exited	<ul style="list-style-type: none"> <li>Before exiting the document <i>After exiting the document</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreQuitDoc (11) <i>Constants.FA_Note_PostQuitDoc (40)</i></li> </ul>
Book exited	<ul style="list-style-type: none"> <li>Before exiting the book <i>After exiting the book</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreQuitBook (24) <i>Constants.FA_Note_PostQuitBook (41)</i></li> </ul>

Event	Notification Point	Notification Constant
First change made to a document since it was opened or saved	After the document is changed	Constants.FA_Note_DirtyDoc (12)
First change made to a book since it was opened or saved	After the book is changed	Constants.FA_Note_DirtyBook (25)
Book saved in Frame binary format	<ul style="list-style-type: none"> <li>• Before saving the book</li> <li>• <i>After saving the book</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreSaveBook (20)</li> <li>• <i>Constants.FA_Note_PostSaveBook (21)</i></li> </ul>
Book saved in MIF format	<ul style="list-style-type: none"> <li>• Before saving the MIF file</li> <li>• <i>After saving the MIF file</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreSaveBookMIF (22)</li> <li>• <i>Constants.FA_Note_PostSaveBookMIF (23)</i></li> </ul>
Document saved with Autosave	<ul style="list-style-type: none"> <li>• Before saving the document</li> <li>• <i>After saving the document</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreAutoSaveDoc (32)</li> <li>• <i>Constants.FA_Note_PostAutoSaveDoc (33)</i></li> </ul>
Document reverted	<ul style="list-style-type: none"> <li>• Before reverting the document</li> <li>• <i>After reverting the document</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreRevertDoc (28)</li> <li>• <i>Constants.FA_Note_PostRevertDoc (29)</i></li> </ul>
Book reverted	<ul style="list-style-type: none"> <li>• Before reverting the book</li> <li>• <i>After reverting the book</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreRevertBook (30)</li> <li>• <i>Constants.FA_Note_PostRevertBook (31)</i></li> </ul>
FrameMaker product exited	<ul style="list-style-type: none"> <li>• Before the OK to Exit dialog box appears</li> <li>• <i>Immediately before exiting the session</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreQuitSession (26)</li> <li>• <i>Constants.FA_Note_PostQuitSession (27)</i></li> </ul>
Another client calls <code>CallClient()</code> with <code>cname</code> set to the current client's name	After the call has been made to <code>CallClient()</code>	Constants.FA_Note_ClientCall (13)

Event	Notification Point	Notification Constant
Any user action, such as a command choice or text entry	After the FrameMaker product finishes processing the user action	Constants.FA_Note_BackToUser (34)
User clicks on Text inset owned by the current client	After the user clicked the inset	Constants.FA_Note_DisplayClientTiDialog (35)
FrameMaker product updates all text insets	When the client needs to update insets that belong to it	Constants.FA_Note_UpdateAllClientTi (36)
FrameMaker product updates a specific text inset	When the client needs to update a specified inset	Constants.FA_Note_UpdateClientTi (37)
Text or graphic imported	<ul style="list-style-type: none"> <li>• Before importing the text or graphic</li> <li style="padding-left: 20px;"><i>After importing the text or graphic</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreImport (38)</li> <li style="padding-left: 20px;"><i>Constants.FA_Note_PostImport (39)</i></li> </ul>
FrameMaker product command invoked or text entered in a document	<ul style="list-style-type: none"> <li>• Before the FrameMaker product executes a command or adds text to the document</li> <li style="padding-left: 20px;"><i>After the FrameMaker product executes a command or adds text to the document</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFunction (42)</li> <li style="padding-left: 20px;"><i>Constants.FA_Note_PostFunction (43)</i></li> </ul>
Mouse button clicked	<ul style="list-style-type: none"> <li>• Before the FrameMaker product responds to the mouse click</li> <li style="padding-left: 20px;"><i>After the FrameMaker product responds to the mouse click</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreMouseCommand (44)</li> <li style="padding-left: 20px;"><i>Constants.FA_Note_PostMouseCommand (45)</i></li> </ul>
Hypertext command invoked	<ul style="list-style-type: none"> <li>• Before the FrameMaker product executes a hypertext command</li> <li style="padding-left: 20px;"><i>After the FrameMaker product executes a hypertext command</i></li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreHypertext (46)</li> <li style="padding-left: 20px;"><i>Constants.FA_Note_PostHypertext (47)</i></li> </ul>

Event	Notification Point	Notification Constant
The user clicked Go To Source in the cross reference dialog box	<ul style="list-style-type: none"> <li>Before the FrameMaker product goes to the cross reference source</li> <li><i>After the FrameMaker product goes to the cross reference source</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreGoToXrefSrc (80)</li> <li><i>Constants.FA_Note_PostGoToXrefSrc (81)</i></li> </ul>
Document or book printed	<ul style="list-style-type: none"> <li>After the user clicks OK in the Print dialog box, but before the FrameMaker product prints the document or book</li> <li><i>After the FrameMaker product prints the document or book</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PrePrint (48)</li> <li><i>Constants.FA_Note_PostPrint (49)</i></li> </ul>
Body page added to a document	After the FrameMaker product adds the body page	Constants.FA_Note_BodyPageAdded (50)
Body page deleted from a document	After the FrameMaker product deletes the body page	Constants.FA_Note_BodyPageDeleted (51)
Structural element inserted	<ul style="list-style-type: none"> <li>Before the element is inserted</li> <li><i>After the element is inserted</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreInsertElement (52)</li> <li><i>Constants.FA_Note_PostInsertElement (53)</i></li> </ul>
Structural element copied	<ul style="list-style-type: none"> <li>Before the element is copied</li> <li><i>After the element is copied</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreCopyElement (60)</li> <li><i>Constants.FA_Note_PostCopyElement (61)</i></li> </ul>
Structural element changed	<ul style="list-style-type: none"> <li>Before the element is changed</li> <li><i>After the element is changed</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreChangeElement (54)</li> <li><i>Constants.FA_Note_PostChangeElement (55)</i></li> </ul>
Structural element wrapped	<ul style="list-style-type: none"> <li>Before the element is wrapped</li> <li><i>After the element is wrapped</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreWrapElement (56)</li> <li><i>Constants.FA_Note_PostWrapElement (57)</i></li> </ul>



Event	Notification Point	Notification Constant
Structural element dragged	<ul style="list-style-type: none"> <li>Before the element is dragged</li> <li><i>After the element is dragged</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreDragElement (58)</li> <li>Constants.FA_Note_PostDragElement (59)</li> </ul>
An attribute value is set	<ul style="list-style-type: none"> <li>Before the attribute value is set</li> <li><i>After the attribute value is set</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreSetAttrValue (62)</li> <li>Constants.FA_Note_PostSetAttrValue (63)</li> </ul>
Element definitions are imported	<ul style="list-style-type: none"> <li>Before the element definitions are imported</li> <li><i>After the element definitions are imported</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreImportElemDefs (64)</li> <li>Constants.FA_Note_PostImportElemDefs (65)</li> </ul>
Inline input of double-byte text	<ul style="list-style-type: none"> <li>Before the text entry</li> <li><i>After the text entry</i></li> </ul>	<ul style="list-style-type: none"> <li>Constants.FA_Note_PreInlineTypeIn (69)</li> <li>Constants.FA_Note_PostInlineTypeIn (70)</li> </ul>
Filter a file on import or export via a filter to file filter	A file to file filter has been invoked—this notification occurs before the file is imported	Constants.FA_Note_FilterFileToFile (75)
Client defined number handling	When the client needs to set the number handling	<p>Constants.FA_Note_RTL_NumberUtility() is used to define the number handling. It has the following values:</p> <ul style="list-style-type: none"> <li>Constants.FV_ITON - Indic to Numeric numbers.</li> <li>Constants.FV_NTOI - Numeric to Indic numbers.</li> <li>Constants.FV_FTON - Farsi to Numeric numbers.</li> <li>Constants.FV_NTOF - Numeric to Farsi numbers.</li> </ul>

The notification constants are numbered sequentially, starting with 0. The method provides a constant, `Constants.FA_Note_Num` (128), that specifies the total number of notifications. This makes it easy to request notification for all notification points.

The method returns `FE_Success` on success. On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.
Constants.FE_BadNotificationNum (-30)	The specified notification number was invalid.

**Returns**

int

**Syntax**

Notification(notification, state)

**Parameters**

Parameter name	Data Type	Optional	Description
notification	int	No	Constant that specifies the notification point.
state	int	No	Specifies whether to turn notification on or off. 1 turns it on, and 0 turns it off

**PopClipboard****Description**

Pops the Clipboard stack, moving the entry on the top of the stack to the Clipboard.

The method returns `FE_Success` on success. On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.
Constants.FE_BadOperation (-27)	Clipboard stack is empty.

**Returns**

int

## Syntax

```
PopClipboard()
```

## Parameters

Parameter name	Data Type	Optional	Description
_	Void	No	NULL

## PrintFAErrno

### Description

Prints the current error status, represented by the global variable, `FA_errno`. The method is useful for debugging clients.

When a method fails, it stores an error code in the global variable, `FA_errno`.

`FA_errno` retains the error code until another function fails and sets it or until your code explicitly sets it.

To determine whether a method call has failed, initialize `FA_errno` to `FE_Success` once before calling the method and check it once after the method call is completed. For example, if you call the `Notification()` method and specify an invalid notification constant, the method sets `FA_errno` to `Constants.FE_BadNotificationNum` (-30). If you subsequently call `PrintFAErrno()`, it prints the string `FE_BadNotificationNum`.

On failure, `PrintFAErrno()` assigns the following value to `FA_errno`:

Error	Reason
<code>Constants.FE_Transport</code> (-1)	A transport error occurred.

### Returns

Void

### Syntax

```
PrintFAErrno(_)
```

## Parameters

Parameter name	Data Type	Optional	Description
_	Void	No	NULL

## PrintImportStatus

### Description

Prints status flags returned by `Import()`. It is useful for debugging your clients.

### Returns

Void

### Syntax

```
PrintImportStatus(p)
```

### Parameters

Parameter name	Data Type	Optional	Description
p	<i>PropVals</i>	No	The property list that <code>Import()</code> returns in <code>importReturnParamspp</code> .

## PrintOpenStatus

### Description

Prints status flags returned by `Open()`. It is useful for debugging your clients.

### Returns

Void

### Syntax

```
PrintOpenStatus(p)
```

## Parameters

Parameter name	Data Type	Optional	Description
p	<i>PropVals</i>	No	The property list that <code>Open ()</code> returns in <code>openReturnParamspp</code> .

## PrintPropVal

### Description

Prints the value of a specified property. It is useful for debugging your clients.

### Returns

Void

### Syntax

```
PrintPropVal (p)
```

### Parameters

Parameter name	Data Type	Optional	Description
p	<i>PropVal</i>	No	The property to print.

## PrintPropVals

### Description

Prints the values in a specified property list. It is useful for debugging your clients.

### Returns

Void

### Syntax

```
PrintPropVals (p)
```

## Parameters

Parameter name	Data Type	Optional	Description
p	<i>PropVals</i>	No	The property to print.

## PrintSaveStatus

### Description

Prints errors returned by `Save()`. It is useful for debugging your clients.

### Returns

Void

### Syntax

```
PrintSaveStatus(p)
```

### Parameters

Parameter name	Data Type	Optional	Description
p	<i>PropVals</i>	No	The property list that <code>Save()</code> returns in <code>saveReturnParamspp</code> .

## PrintTextItem

### Description

Prints the text in a specified text item. It is useful for debugging clients.

### Returns

Void

### Syntax

```
PrintTextItem(textItem)
```

## Parameters

Parameter name	Data Type	Optional	Description
textItem	<i>TextItem</i>	No	The text item to print.

## PrintTextItems

### Description

Prints the text in a specified set of text items (`TextItems` structure). It is useful for debugging clients.

### Returns

Void

### Syntax

```
PrintTextItems (textItems)
```

### Parameters

Parameter name	Data Type	Optional	Description
textItems	<i>TextItems</i>	No	The set of text items to print.

## PrintUpdateBookStatus

### Description

Prints errors returned by `UpdateBook()`. It is useful for debugging your clients.

### Returns

Void

### Syntax

```
PrintUpdateBookStatus (p)
```

## Parameters

Parameter name	Data Type	Optional	Description
p	<i>PropVals</i>	No	The property list that <code>UpdateBook()</code> returns in <code>updateReturnParamspp</code> .

## PushClipboard

### Description

Pushes the current Clipboard contents onto the Clipboard stack. The method is useful if you want to use Clipboard functions, such as `Copy()` or `Cut()`, without losing the Clipboard's original contents.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

### Returns

Void

### Syntax

```
PushClipboard()
```

### Parameters

Parameter name	Data Type	Optional	Description
	Void	No	NULL

## ReturnValue

### Description

Sets a return value for a client-defined callback. The method allows a client to provide status information to the FrameMaker product or client that called the callback. You can call this method in the following callbacks:

- `DialogEvent()`



- Notify()

ReturnValue() is useful for canceling FrameMaker product operations. When your client receives a Constants.FA\_PreNotificationPoint notification for an operation, it can cancel the operation by calling ReturnValue() with retvalue set to Constants.FR\_CancelOperation (-10001).

For example, if your client's Notify() callback responds to all Constants.FA\_Note\_PrePrint (48) notifications by calling ReturnValue() with retvalue set to Constants.FR\_CancelOperation (-10001), FrameMaker cancels all print operations.

Your client can also call ReturnValue() in a DialogEvent() callback to prevent FrameMaker from closing a modal dialog box.

You can set retvalue to any integer. If you client sets retval in response to one of the following notifications, it can use the following listed constants:

Notification	Values that client can pass to ReturnValue()	Meaning
Constants.FA_NOTE_PreNotificationPoint	Constants.FR_CancelOperation (-10001)	Cancel the operation for which the notification was issued
Constants.FA_Note_ClientCall (13)	Any value recognized by the client that called CallClient().	Client-defined.
Constants.FA_Note_FilterIn (14)	<ul style="list-style-type: none"> <li>• The document into which the file was filtered.</li> <li>0</li> </ul>	<ul style="list-style-type: none"> <li>• The document was filtered successfully.</li> <li><i>The document was not filtered successfully.</i></li> </ul>
Constants.FA_Note_DisplayClientTiDialog (35)	Constants.FR_DisplayedTiDialog (-10002)	The client has displayed its version of the Text Inset Properties dialog box.
Constants.FA_Note_PreSaveAsPDFDialog (71)	<ul style="list-style-type: none"> <li>• Constants.FR_CancelOperation (-10001)</li> <li>Constants.FR_SkipStep (-10004)</li> </ul>	<ul style="list-style-type: none"> <li>• Cancel the Save As PDF operation.</li> <li><i>Do not display the Acrobat Settings dialog box</i></li> </ul>
Constants.FA_Note_PostSaveAsPDFDialog (72)	Constants.FR_CancelOperation (-10001)	Cancel the Save as PDF operation. <b>NOTE:</b> <i>This event occurs before the distilling operation begins. You can now cancel the operation after the user closes Save As PDF dialog box.</i>
Constants.FA_Note_PreDistill (73)	Constants.FR_CancelOperation (-10001)	Cancel the Save as PDF operation.

Calling `ReturnValue()` for notifications other than those listed above, has no effect.

A client can also call `ReturnValue()` in a `DialogEvent()` callback that responds to actions in a client-defined modal dialog box.

Normally, when the user clicks a button in a client-defined modal dialog box, `FrameMaker` calls the client's `DialogEvent()` callback and then closes the dialog box. However, if the client's `DialogEvent()` callback calls `ReturnValue()` with `retvalue` set to `Constants.FR_DialogStayUp (-10000)`, `FrameMaker` does not close the dialog box.

The following table lists the values that a client can pass to `ReturnValue()` in a `DialogEvent()` callback:

Value	Meaning
<code>Constants.FR_DialogStayUp (-10000)</code>	Do not close the modal dialog box in which the event occurred.
Any other value	Close the modal dialog box.

On success, `ReturnValue()` returns the value of the `retvalue` parameter that was set, the previous time `ReturnValue()` was called in the current callback function.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
<code>Constants.FE_Transport (-1)</code>	A transport error occurred.

## Returns

`int`

## Syntax

`ReturnValue(retvalue)`

## Parameters

Parameter name	Data Type	Optional	Description
<code>retvalue</code>	<code>int</code>	No	The value to return.

## ShutDown

### Description

Closes an ESTK client's connection.

### Returns

Void

### Syntax

```
ShutDown ( _ )
```

### Parameters

Parameter name	Data Type	Optional	Description
_	Void	No	NULL

## SimpleNewDoc

### Description

Creates a new document from a specified template.

***NOTE:** If you call `SimpleNewDoc()` with `interactive` set to `True` and the user clicks either `Portrait`, `Custom`, or `Landscape` in the `New` dialog box, `SimpleNewDoc()` does not create a document. Instead, it returns `0` and sets `FA_errno` to either `Constants.FE_WantsPortrait` (`-48`), `Constants.FE_WantsCustom` (`-46`), or `Constants.FE_WantsLandscape` (`-47`), respectively. It is up to your client to create a portrait, custom, or landscape document. **NOTE:** For more information on creating custom documents, see [CustomDoc](#).*

If `SimpleNewDoc()` fails, the method assigns an error code to `FA_errno`. The error codes for `SimpleNewDoc()` are the same as those for `Open()`. For a list of these error codes, see [Open](#).

### Returns

[Doc](#)

### Syntax

```
SimpleNewDoc(templateName, interactive)
```

## Parameters

Parameter name	Data Type	Optional	Description
templateName	string	No	The absolute pathname of the template to use.
interactive	int	No	Specifies whether FrameMaker displays messages and warnings to the user.

## UserCancel

### Description

Determines whether the user has chosen the Cancel command (ESC) since the current callback function was called.

`UserCancel()` is useful for clients that conduct extensive processing that the user may want to cancel. For example, if your client processes all the documents in a book, it can call `UserCancel()` after it processes each document.

If `UserCancel()` returns `True`, your client can abort the processing.

Returns `True` if the user has executed the Cancel gesture, or `False` if the user has not executed the Cancel gesture.

### Returns

int

### Syntax

`UserCancel()`

### Parameters

Parameter name	Data Type	Optional	Description
	Void	No	NULL

## SimpleOpen

### Description

Opens a document or book.

If you call `SimpleOpen()` with `interactive` set to `True`, FrameMaker displays the Open dialog box. It uses the path specified by the session property, `OpenDir`, as the default path.

If a warning or error condition arises, FrameMaker notifies the user. For example, if a document uses fonts that are not available, FrameMaker displays a dialog box that allows the user to cancel the operation or to continue and remap the fonts. If you set `interactive` to `False`, FrameMaker does not display the Open dialog box or other messages and warnings. If it is necessary to modify a file to continue opening it, `SimpleOpen()` aborts the operation without notifying the user, and returns 0.

**NOTE:** You cannot use `SimpleOpen()` to open filterable files. To open filterable files, use [Open](#)

The method returns the document opened, on success.

If `SimpleOpen()` fails, the method assigns an error code to `FA_errno`. The error codes for `SimpleOpen()` are the same as those for [Open](#).

### Returns

Object

### Syntax

```
SimpleOpen(fileName, interactive)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>fileName</code>	<code>string</code>	No	The absolute pathname of the file to open.
<code>interactive</code>	<code>int</code>	No	Specifies whether the FrameMaker product displays messages and warnings to the user. 1 instructs the FrameMaker product to display messages and warnings.

## UpdateMenus

### Description

Updates the top order menu set. For example, when a new menu file is read, or when a view only, book menu, quick menu or similar is loaded, this updates the newly read menus (in case of menu file) or different top menu bar (in case of view only, book and quick menus).

### Returns

Void

## Syntax

```
UpdateMenus()
```

## Parameters

Parameter name	Data Type	Optional	Description
	Void	No	NULL

## AddMenu

### Description

Adds a menu to a specified menu.

The method returns `FE_Success` on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Invalid product.
<code>Constants.FE_BadOperation (-27)</code>	One of: <ul style="list-style-type: none"> <li><i>menu</i> is null or empty</li> <li><i>tomenu</i> is null or empty</li> <li><i>value of menu</i> is already defined as a command</li> </ul>
<code>Constants.FE_BadParameter (-43)</code>	One of: <ul style="list-style-type: none"> <li><i>tomenu</i> <i>and</i> <i>menu</i> have the same values</li> <li><i>tomenu</i> is not previously defined</li> <li><i>tomenu</i> is not a menu</li> </ul>

### Returns

```
int
```

### Syntax

```
AddMenu(toMenu, menu, label)
```

## Parameters

Parameter name	Data Type	Optional	Description
toMenu	string	No	The menu tag to which the menu is to be added.
menu	string	No	The new menu which is being added to the existing menu.
label	string	No	The name by which the newly added menu is visible.

## RemoveMenu

### Description

Deletes the specified menu.

The method returns `FE_Success` on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Invalid product.

### Returns

int

### Syntax

```
RemoveMenu(fromMenu, label)
```

### Parameters

Parameter name	Data Type	Optional	Description
fromMenu	string	No	The parent menu from which the menu is to be deleted.
label	string	No	The menu to delete.

## MenuExists

### Description

Indicates whether a specified menu exists.

The method returns 1 if the menu exists or 0 if the menu does not exist.

### Returns

int

### Syntax

```
MenuExists(menu)
```

### Parameters

Parameter name	Data Type	Optional	Description
menu	string	No	The menu to be checked for existence.

## ShortcutExists

### Description

Indicates whether a specified command exists.

The method returns 1 if the command exists or 0 if the command does not exist.

### Returns

int

### Syntax

```
ShortcutExists(shortcut)
```

### Parameters

Parameter name	Data Type	Optional	Description
shortcut	string	No	The command to be checked for existence.



## GetExportDefaultParams

### Description

Returns the PropVals structure containing the default export parameters.

### Returns

*PropVals*

### Syntax

```
GetExportDefaultParams ()
```

### Parameters

Parameter name	Data Type	Optional	Description
	Void	No	Null

## PrintExportStatus

### Description

### Returns

Void

### Syntax

```
PrintExportStatus (p)
```

### Parameters

Parameter name	Data Type	Optional	Description
p	<i>PropVals</i>	No	

## ErrorName

### Description

Returns the string corresponding to the value of `FA_errno`.

**Returns**`string`**Syntax**`ErrorMessage(i)`**Parameters**

Parameter name	Data Type	Optional	Description
<code>i</code>	<code>int</code>	No	

**PrintErrno****Description**

Prints the value of `FA_errno` to the console.

**Returns**`Void`**Syntax**`PrintErrno(i)`**Parameters**

Parameter name	Data Type	Optional	Description
<code>i</code>	<code>int</code>	No	

**NotificationName****Description****Returns**`string`**Syntax**`NotificationName(i)`

## Parameters

Parameter name	Data Type	Optional	Description
i	int	No	

## AllocatePropVals

### Description

Allocates memory for a property list.

On success, the method returns a property list (a `PropVals` data structure). The returned `PropVals` structure references memory that is allocated by the method. If `AllocatePropVals()` fails, the method sets the `len` field of the returned structure to 0.

### Returns

*PropVals*

### Syntax

`AllocatePropVals (numProps)`

## Parameters

Parameter name	Data Type	Optional	Description
numProps	int	No	The number of properties in the property list.

## Open

### Description

Opens a document or book. It can also create a new document.

`Open()` allows you to specify a property list telling `FrameMaker` how to open or create the file and how to deal with error and warning conditions.

For example, you can specify whether to abort or to continue opening a document if it contains fonts that are not available. If the file is already open and invisible, it will make the file visible.

To get a property list to specify for the `openParams` parameter, use `GetOpenDefaultParams()` or create the list from scratch.

To create a new document with `Open()`, set the `Constants.FS_NewDoc (29)` property in the `openParams` property list to `True`.

**NOTE:** When creating a new document (`Constants.FS_NewDoc (29)` is `True`) and you display the New dialog box (`Constants.FS_ShowBrowser (1)` is `True`) and if the user clicks either Portrait, Custom, or Landscape in the New dialog box, `Open()` does not create a document. Instead, it returns 0 and sets `FA_errno` to either `Constants.FE_WantsPortrait (-48)`, `Constants.FE_WantsCustom (-46)`, or `Constants.FE_WantsLandscape (-47)`, respectively. It is up to your client to create a portrait, custom, or landscape document. **NOTE:** For more information on creating custom documents, see [CustomDoc](#).

The method returns the document or book if it opens it successfully, or 0 if an error occurs.

The property list to which `openReturnParams` is set has the properties shown in the following table:

Property	Meaning and possible values
<code>Constants.FS_OpenedFileName (1)</code>	A string that specifies the opened file's pathname. If you scripted <code>Constants.FS_ShowBrowser (1)</code> , or the file was filtered, or you did not specify the pathname, this pathname can be different from the one you specified in the Open script.
<code>Constants.FS_OpenNativeError (2)</code>	The error condition; normally the same value as <code>FA_errno</code> . If the file is opened successfully, it is set to <code>FE_Success</code> .
<code>Constants.FS_OpenStatus (3)</code>	A bit field indicating what happened when the file was opened.

Both the `Constants.FS_OpenNativeError (2)` property and the `FA_errno` global variable indicate the result of a call to `Open()`. The following table lists the possible status flags and the `FA_errno` and `Constants.FS_OpenNativeError (2)` values associated with them.



FS_OpenNativeError and FA_errno values	Possible FS_OpenStatus flags
Constants.FE_Success (0) (file was opened)	<ul style="list-style-type: none"> <li>• Constants.FV_FileHasNewName (32) - Filename was changed from the name specified in the <i>Open ()</i> call.</li> <li><i>Constants.FV_RecoverFileUsed (33) - Recover file was present, and it was used.</i></li> <li><i>Constants.FV_AutoSaveFileUsed (34) - Autosave file was present, and the user or the <i>Open ()</i> script chose to use it.</i></li> <li><i>Constants.FV_FileWasFiltered (35) - File was filterable and it was filtered.</i></li> <li><i>Constants.FV_FontsWereMapped (36) - The document contained unavailable fonts, which were mapped to substitute fonts.</i></li> <li><i>Constants.FV_FontMetricsChanged (37) - The file contained fonts with changed metrics, but it was opened anyway.</i></li> <li><i>Constants.FV_FontsMappedInCatalog (38) - The Paragraph or Character Catalog used unavailable fonts, which were mapped to substitute fonts.</i></li> <li><i>Constants.FV_LanguagesWerentFound (39) - The document used some unavailable languages, but it was opened anyway.</i></li> <li><i>Constants.FV_BeefyDoc (40) - The document file was extremely large, but it was opened anyway.</i></li> <li><i>Constants.FV_FilesOldVersion (41) - The file was from an old FrameMaker product version, but the user or the <i>Open ()</i> script chose to open it anyway.</i></li> <li><i>Constants.FV_FileStructureStripped (42) - The file had FrameMaker features, which the user or the <i>Open ()</i> script chose to strip.</i></li> <li><i>Constants.FV_FilesText (43) - The file was a Text Only file, but the user or the <i>Open ()</i> script chose to open it anyway.</i></li> <li><i>Constants.FV_OpenedViewOnly (44) - The user or the <i>Open ()</i> script chose to open the file as a View Only file.</i></li> <li><i>Constants.FV_EditableCopyOpened(45) - The file was in use and the user or the <i>Open ()</i> script opened an editable copy.</i></li> <li><i>Constants.FV_BadFileRefsWereMapped (46) - File reference contained illegal characters; the illegal characters were converted to something safe.</i></li> <li><i>Constants.FV_ReferencedFilesWerentFound (47) - Imported graphics files could not be found, but the file was opened anyway.</i></li> </ul>

FS_OpenNativeError and FA_errno values	Possible FS_OpenStatus flags
Constants.FE_Success (0) (file was opened)	<ul style="list-style-type: none"> <li>Constants.FV_FileAlreadyOpen (48) - The file was in use and the user or the <i>Open()</i> script opened another copy.</li> <li><i>Constants.FV_UnresolvedXRefs (49) - There were unresolved cross-references, but the file was opened anyway.</i></li> <li><i>Constants.FV_UnresolvedTextInsets(50) - There were unresolved text insets, but the file was opened anyway.</i></li> </ul>
Constants.FE_SystemError (-42) (file was not opened)	<ul style="list-style-type: none"> <li>Constants.FV_TooManyWindows(128) - Too many windows were open.</li> <li><i>Constants.FV_BadTemplate (129) - A bad template was specified.</i></li> <li><i>Constants.FV_FileNotReadable (130) - Do not have read permission for the file.</i></li> </ul>

FS_OpenNativeError and FA_errno values	Possible FS_OpenStatus flags
Constants.FE_BadParameter (-43) (file was not opened)	<ul style="list-style-type: none"> <li>• Constants.FV_FileHadStructure(64) - File had FrameMaker features, but current FrameMaker product is not FrameMaker. <i>Constants.FV_FileAlreadyOpenThisSession (65) - File is already open and script disallowed opening another copy.</i> <i>Constants.FV_BadFileType (66) - File was an executable file or other unreadable type.</i> <i>Constants.FV_BadFileName (67) - Specified filename was invalid.</i> <i>Constants.FV_CantNewBooks (68) - Script specified a book that did not exist (the Open operation cannot create a new book).</i> <i>Constants.FV_BadScriptValue (70) - Open () script contained an invalid property value.</i> <i>Constants.FV_MissingScript (71) - Open () was called without a script.</i> <i>Constants.FV_CantForceOpenAsText (72) - Open () script attempted to open the file as text, but file was of the wrong type.</i> <i>Constants.FV_DisallowedType (73) - File was a Frame binary document and the Open () script disallowed it.</i> <i>Constants.FV_DocDamagedByTextFilter (74) - File was a text document and was damaged when it was filtered.</i> <i>Constants.FV_DocHeadersDamaged (75) - The document headers were damaged (probably because of a file system problem).</i> <i>Constants.FV_DocWrongSize (76) - File is the wrong size (probably because of a file system problem).</i> <i>Constants.FV_ChecksumDamage (77) - Bad checksum.</i></li> </ul>



FS_OpenNativeError and FA_errno values	Possible FS_OpenStatus flags
Constants.FE_Canceled (-44) (file was not opened)	<ul style="list-style-type: none"> <li>• Constants.FV_CancelUseRecoverFile (96) - A recover file was present, so the user or the <i>Open ()</i> script canceled the Open operation.</li> <li><i>Constants.FV_CancelUseAutoSaveFile (97) - An Autosave file was present, so the user or the <i>Open ()</i> script canceled the Open operation.</i></li> <li><i>Constants.FV_CancelFileIsText (98) - The file was text, so the user or the <i>Open ()</i> script canceled the Open operation.</i></li> <li><i>Constants.FV_CancelFileIsInUse (99) - The file was in use, so the user or the <i>Open ()</i> script canceled the Open operation.</i></li> <li><i>Constants.FV_CancelFileHasStructure (100) - The file had structure, so the user or the script canceled the Open operation.</i></li> <li><i>Constants.FV_CancelReferencedFilesNotFound (101) - The file contained referenced files that were not available, so the user or the <i>Open ()</i> script canceled the Open operation.</i></li> <li><i>Constants.FV_CancelLanguagesNotFound (102) - The file contained languages that were not available, so the user or the <i>Open ()</i> script canceled the Open operation.</i></li> <li><i>Constants.FV_CancelFontsMapped (103) - The document contained fonts that needed to be mapped to other fonts, so the user or the <i>Open ()</i> script canceled the Open operation.</i></li> <li><i>Constants.FV_CancelFontMetricsChanged (104) - The file contained fonts with changed metrics, so the user or the <i>Open ()</i> script canceled the Open operation.</i></li> <li><i>Constants.FV_CancelFontsMappedInCatalog (105) - The document's Character Catalog or Paragraph Catalog contained fonts that needed to be mapped to other fonts, so the user or the <i>Open ()</i> script canceled the Open operation.</i></li> <li><i>Constants.FV_CancelFileIsDoc (106) - The file was a document and the <i>Open ()</i> script disallowed it.</i></li> <li><i>Constants.FV_CancelFileIsMIF (107) - The file was a MIF file and the <i>Open ()</i> script disallowed it.</i></li> <li><i>Constants.FV_CancelBook (108) - The file was a book and the <i>Open ()</i> script disallowed it.</i></li> <li><i>Constants.FV_CancelBookMIF (109) - The file was a MIF file and the <i>Open ()</i> script disallowed it.</i></li> <li><i>Constants.FV_CancelFileIsFilterable (110) - The file was a filterable file and the <i>Open ()</i> script disallowed it.</i></li> <li><i>Constants.FV_CancelFileIsOldVersion (111) - The file was from an old version of a FrameMaker product, so the user or the <i>Open ()</i> script canceled the Open operation.</i></li> </ul>
	<ul style="list-style-type: none"> <li>• Constants.FV_UserCanceled (112) - The user canceled the Open operation.</li> </ul>
	<ul style="list-style-type: none"> <li>• Constants.FV_CancelFileBrowser (113) - The user canceled</li> </ul>

FS_OpenNativeError and FA_errno values	Possible FS_OpenStatus flags
<ul style="list-style-type: none"> <li>Constants.FE_Success (0)</li> <li>Constants.FE_Canceled (-44)</li> <li>Constants.FE_FailedState (-45)</li> <li>Constants.FE_CanceledByClient (-86)</li> </ul>	<ul style="list-style-type: none"> <li>Constants.FV_LockWasReset (0) - File lock was reset.</li> <li>Constants.FV_LockNotReset (1) - File had a lock that was not reset.</li> <li>Constants.FV_LockCouldntBeReset (2) - File had a lock that could not be reset.</li> <li>Constants.FV_FileWasInUse (3) - File was in use.</li> <li>Constants.FV_FileIsViewOnly (4) - File is a View Only file.</li> <li>Constants.FV_LockWasInvalid (5) - File had an invalid lock.</li> <li>Constants.FV_FileIsNotWritable (6) - The file was not writable, and the user canceled the open through the alert.</li> <li>Constants.FV_FileModDateChanged (7) - The file has changed since the last time it was opened or saved in the current session.</li> </ul>

To determine if a particular FS\_OpenStatus bit is set, use [CheckStatus](#).

### Returns

Object

### Syntax

Open(fileName, openParams, openReturnParams)

### Parameters

Parameter name	Data Type	Optional	Description
fileName	string	No	The full pathname of the file to open. If you are using Open () to create a document, specify the name of the template to use.
openParams	<a href="#">PropVals</a>	No	A property list telling the FrameMaker product how to open the file and how to respond to errors and other conditions. To use the default list, specify NULL.
openReturnParams	<a href="#">PropVals</a>	No	A property list that returns the filename and provides information about how FrameMaker opened the file. It must be initialized before you call Open () .

## ScrollBar

### Description

Displays an array of items and allows the user to choose one.

*NOTE: If you set `default` to -1, always check to make sure the value returned in `selected_item` is 0 or greater before you use it as an array index. If you set `default` to -1 and the user clicks OK without choosing an item, the value returned in `selected_item` will be -1.*

Returns 0 if the user clicked OK, or a nonzero value if the user clicked Cancel or an error occurred.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	The user clicked Cancel, or a transport error occurred.

### Returns

int

### Syntax

```
ScrollBar(title, stringslist, _default)
```

### Parameters

Parameter name	Data Type	Optional	Description
title	string	No	The title that appears on the dialog box.
stringslist	<i>Strings</i>	No	The list of items to appear in the scroll list.
_default	int	No	The index of the item that is selected when the dialog box first appears. For no default, specify -1.

## ChooseFile

### Description

Displays dialog boxes similar to a FrameMaker's Open and Save dialog boxes. The method displays directories and files in a scroll list and allows the user to choose a file or directory.

You can set the following values for `mode`:

Mode Constant	Dialog Box Type
Constants.FV_ChoseSelect (0)	Dialog box that allows the user to choose a file by clicking Select. It provides an input field into which the user can type a filename.
Constants.FV_ChoseOpen (1)	Dialog box that allows the user to choose a file by clicking Open. It provides an input field into which the user can type a filename.
Constants.FV_ChoseSave (2)	Dialog box that allows the user to save a file. It provides Save and Cancel buttons and an input field.
Constants.FV_ChoseOpenDir (3)	Dialog box that allows the user to choose a directory.

The method returns 0 if the user clicked Open, Select, Use, or Save; a nonzero value if the user clicked Cancel or an error occurred.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

## Returns

`int`

## Syntax

```
ChooseFile(title, directory, stuffVal, mode)
```

## Parameters

Parameter name	Data Type	Optional	Description
<code>title</code>	<code>string</code>	No	The message that appears in the dialog box.
<code>directory</code>	<code>string</code>	No	The default directory when the dialog box is first displayed. If you specify an empty string, the last directory used by an ESTK client is used. If no ESTK client has used a directory, the directory specified by the session property, <code>OpenDir</code> , is used.

Parameter name	Data Type	Optional	Description
stuffVal	string	No	The default value that appears in the input field when the dialog box first appears. If the dialog box type specified by mode does not have an input field, this string is ignored.
mode	int	No	A constant specifying the type of dialog box.

## PromptInt

### Description

Displays a dialog box that prompts the user for a single integer value. It allows you to provide a default value, which appears in the entry field when the dialog box appears. The dialog box has OK and Cancel buttons.

`PromptInt()` does not assign a value to `*intp` if the user clicks Cancel. If the user types alphabetic text after a number, the method ignores the text and just returns the value of the number.

The method returns 0 if the user clicked OK; a nonzero value if the user clicked Cancel or an error occurred.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

### Returns

int

### Syntax

```
PromptInt(message, stuffVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
message	string	No	The message that appears in the dialog box. It must be 255 characters or less.

Parameter name	Data Type	Optional	Description
stuffVal	string	No	The default value that appears in the input field when the dialog box is first displayed.

## PromptMetric

### Description

Displays a dialog box that prompts the user for a single metric value. It allows you to provide a default value, which appears in the entry field when the dialog box appears. The dialog box has OK and Cancel buttons.

`PromptMetric()` does not assign a value to `*metricp` if the user clicks Cancel.

`PromptMetric()` dialog boxes behave like metric dialog boxes in the user interface. If the user types a number followed by a string that represents a unit (for example 10pts or 5"), the method converts the number into the equivalent number of metric units. If the user does not specify a unit, the method uses points (metric 65536).

The method returns 0 if the user clicked OK; a nonzero value if the user clicked Cancel or an error occurred.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

### Returns

int

### Syntax

```
PromptMetric(message, stuffVal, defaultunit)
```

### Parameters

Parameter name	Data Type	Optional	Description
message	string	No	The message that appears in the dialog box. It must be 255 characters or less.

Parameter name	Data Type	Optional	Description
stuffVal	string	No	The default value that appears in the input field when the dialog box is first displayed.
defaultunit	int	No	The metric unit to use if the user does not specify one.

## PromptString

### Description

Displays a dialog box that prompts the user for a single string value. It allows you to provide a default string, which appears in the entry field when the dialog box appears. The dialog box has OK and Cancel buttons.

**NOTE:** Do not call `PromptString()` to prompt the user for a pathname. If the user enters a pathname as a string, the backslash character (`\`) is interpreted as a special escape character. For example, the characters `\s` represent a space. If the user enters the pathname `c:\sample`, this string is interpreted as `c:ample`. To prompt the user for a pathname, use [ChooseFile](#) to display a file selection dialog box.

The method returns 0 if the user clicked OK; a nonzero value if the user clicked Cancel or an error occurred.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

### Returns

string

### Syntax

```
PromptString(message, stuffVal)
```

## Parameters

Parameter name	Data Type	Optional	Description
message	string	No	The message that appears in the dialog box. It must be 255 characters or less. Newline and linefeed characters are ignored.
stuffVal	string	No	The default value that appears in the input field when the dialog box is first displayed.

## Service

### Description

Services calls from FrameMaker to the ESTK API.

`Service()` waits for and dispatches a single Windows message. If your application contains its own message processing loop you need not call this routine.

It is not feasible for console applications to receive ESTK notifications since there is no way for them to both wait for user input and call `Service()`. Consequently, console applications should not register for notifications.

The method returns the number of bits the call to `select()` selected, if any.

### Returns

int

### Syntax

`Service()`

## FileScrollBox

### Description

Displays a file scrollbar using the given title and the given directory as the path.

### Returns

int

### Syntax

`FileScrollBox(title, directory)`



## Parameters

Parameter name	Data Type	Optional	Description
title	string	No	The title to display on the scroll box.
directory	string	No	The directory path to display.

## SetCurrentWorkspace

### Description

Sets the current workspace to the specified string.

### Returns

null

### Syntax

```
var str="Review";
SetCurrentWorkspace(str);
```

### Parameters

Parameter name	Data Type	Optional	Description
str	string	No	The name of the workspace to set.

## GetWorkspaceName

### Description

Gets the name of the current workspace.

### Returns

string

### Syntax

```
var str=GetWorkspaceName();
```



# CMS Connector Framework

## CMS API Data Structures and Enum Constants

### CMSResult

#### CMSResult

##### Description

Specifies the state of a Command's result for CMSCommand.

##### Returns

CMSResult

##### Syntax

```
CMSResult (status, opResult, message, cmsItems) ;
```

##### Parameters

Parameter name	Data Type	Optional	Description
status	int	No	Command's status
opResult	int	No	Operation's result. If CMS Command needs CMSTree update, assign CMSOpResult values (See following enum),else can return any value depending on operation. For example: opResult = True/False for FA_CMSIsValidCommand , opResult = CMSPropertyNewMaxOpCode for FA_CMSGetPropertyMaxOpCode
message	string	No	If operation fails, user can send error message to FrameMaker.For FA_CMSObjectOpenReadOnly, FA_CMSObjectEdit command, user can return file-name which is downloaded.
cmsItems	CMSObjects	No	List of CMS object

The possible values of the CMSResult.opResult field are:

opResult constant	Meaning
Constants.CMSOpNone	None
Constants.CMSOpItemUpdated	CMS item is updated
Constants.CMSOpDependentsUpdated	Dependents are updated
Constants.CMSOpDependentsDeleted	Dependents are deleted
Constants.CMSOpItemDeleted	CMS item is deleted
Constants.CMSOpChildAdded	Child is added
Constants.CMSOpRootUpdated	Root is updated

If CMS SetProperty() fails, the API assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_CMSRootObjectExists	The API tries to set a root which already exists.
Constants.FE_CMSBadItemFileType	The file type expected by the cms object does not match the valid file type.
Constants.FE_CMSBadItemType	The item type expected by the cms object does not match the valid item type
Constants.FE_CMSBadItemContainerType	The container value expected by the cms object is not properly set

## CMSItemFileType

### Description

Enum constants used to determine File-Type of a CMS Object.

The possible values of the Constants.FP\_CMSItemProperty\_ItemFileType fields are:

File Type constant	Meaning
--------------------	---------

Constants.FV_CMSItemFileTypeValue_Xml	File type is XML
Constants.FV_CMSItemFileTypeValue_FmDoc	File type is FM
Constants.FV_CMSItemFileTypeValue_Mif	File type is MIF
Constants.FV_CMSItemFileTypeValue_DitaMap	File type is DITAMAP
Constants.FV_CMSItemFileTypeValue_DitaTopic	File type is DITA topic
Constants.FV_CMSItemFileTypeValue_FmBook	File type is Book
Constants.FV_CMSItemFileTypeValue_Text	File type is text
Constants.FV_CMSItemFileTypeValue_Img	File type is image
Constants.FV_CMSItemFileTypeValue_General	File type is General

## CMSProperties

### Description

A `CMSProperties` structure specifies a set of CMS object properties.

## CMSMenuItem

### Description

The `CMSMenuItem` structure describes a custom menu definition.

This structure is used for creating a custom menu entry in the context menu available in CMS tree and CMS dialogs.

## CMSCheckinParam

### Description

The `CMSCheckinParam` structure describes the checkin parameter.

This structure is returned by `CMSShowCheckinUI` API for getting all the user interface state after user accepts the dialog changes by pressing OK button.

## CMSDeleteParam

### Description

The `CMSDeleteParam` structure describes the delete parameter.

This structure is returned by `CMSShowDeleteUI` API for getting all the user interface state after the user accepts the dialog changes by clicking OK.

## CMSInfo

### Description

The `CMSInfo` structure describes a single CMS registration information definition

## CMSInfos

### Description

The `CMSInfos` structure specifies a set of CMS registration information.

## Error Codes

If the CMS API encounters an error condition, the API assigns an error code to the global variable `FA_errno`. The following table lists the error codes and their meanings. Error codes are also listed in the `fcmsapi.h` header file.

Error code	Meaning
<code>Constants.FE_CMSNameAlreadyRegistered</code>	The API attempted to register a CMS that is already registered.
<code>Constants.FE_CMSBadSessionId</code>	The client specified an invalid session ID.
<code>Constants.FE_CMSBadObjectId</code>	The client specified an invalid CMS object ID.
<code>Constants.FE_CMSSessionFailed</code>	The client failed to create a session.
<code>Constants.CMSBadCommandId</code>	The client specified an invalid command ID.
<code>Constants.FE_CMSObjectCreationFailed</code>	The <code>CMSCreateObject</code> API fails to create a CMS object.
<code>Constants.FE_CMSRootObjectExists</code>	The API tried to set a root that already exists.
<code>Constants.FE_CMSBadItemFileType</code>	The file type expected by the CMS object does not match the valid file type.

Constants.FE_CMSBadItemType	The item type expected by the CMS object does not match the valid item type.
Constants.FE_CMSBadItemContainerType	The container value expected by the cms object is not properly set
Constants.FE_CMSSessionCreationFailed	If Session creation fails, set status to this value.
Constants.FE_CMSIsValidCMSCommand	If user wants FrameMaker to take care of IsValidCMSCommand, set opResult to this value.
Constants.FE_CMSFailedLogin	The CMSLogin API fails to log into a CMS.
Constants.FE_CMSFailedLogout	The CMSLogout API fails to log out from a CMS.
Constants.FE_CMSFailedCheckout	The CMSCheckout API failed to checkout a file.
Constants.FE_CMSFailedCheckin	The CMSCheckin API failed to check in a file
Constants.FE_CMSFailedCancelCheckout	The CMSCancelCheckout API fails to cancelcheckout of a file.
Constants.FE_CMSFailedDelete	The CMSDelete API failed to delete a cms object
Constants.FE_CMSFailedOpenFile	The CMSOpenFile API failed to open a file
Constants.FE_CMSFailedUploadObject	The CMSUploadObject API failed to upload afile or folder.
Constants.FE_CMSFailedDownloadObject	The CMSzDownloadObject API failed to download a file.
Constants.FE_CMSFailedGetItemFrompath	The GetCMSObjectFromPath API failed to return a CMS object from server path.

## CMS API functions

A list of the prominent CMS API functions:

[\*CMSRegister\*](#)

[\*CMSCreateObject\*](#)

[\*CMSEnableCommand\(\)\*](#)

[\*CMSDisableCommand\(\)\*](#)

[\*CMSAddMenuEntry\*](#)

[\*CMSGetCmsIdFromName\*](#)

[\*CMSGetCMSInfo\(\)\*](#)

[\*CMSGetCmsIdFromSession\*](#)

*CMSShowConfigLoginUI()*  
*CMSShowCheckoutUI()*  
*CMSShowCheckinUI()*  
*CMSShowCancelCheckoutUI()*  
*CMSShowDeleteUI()*  
*CMSShowCommonListUI()*  
*CMSShowPropertyUI()*  
*CMSShowPropertyUIWithTitle*  
*CMSGetProperty()*  
*CMSGetProperties()*  
*CMSSetProperty()*  
*CMSSetProperties()*

## APIs to automate CMS connector functionality

### **CMSLogin**

#### **Description**

Logs into a particular CMS based on the connection details

#### **Returns**

Returns the handle of the new CMS connection if the operation is successful. Else sets FA\_errno to Constants.FE\_CMSErrordLogin.

#### **Syntax**

```
CMSLogin (setVal);
```



## Parameters

Parameter name	Data Type	Optional	Description
setVal	IdValue Pairs	No	Id value pairs to specify the connection parameter. The valid Ids are: Constants.FV_CMSCCommandNameId - Name of the connection Constants.FV_CMSCCommandConnTypeId -Connection Type Constants.FV_CMSCCommandServerId -Server Name Constants.FV_CMSCCommandUserNameId -User Name Constants.FV_CMSCCommandPasswordId-Password Constants.FV_CMSCCommandUserField1-Optional User Field1 Constants.FV_CMSCCommandRepositoryId-Repository name for documentum Constants.FV_CMSCCommandUserField2-Optional User Field2

## CMSLogout

### Description

Logs out the user from a particular CMS connection

### Returns

Returns `FE_Success` if the operation is successful, else sets `FA_erno` to `Constants.FE_CMSErrordLogout`

### Syntax

```
CMSLogout (cmsSessionId) ;
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session

## CMSCheckout

### Description

Checks out a file from the CMS

### Returns

Returns `FE_Success` if the operation is successful. Else sets `FA_errno` to `Constants.FE_CMSFailedCheckout`

### Syntax

```
CMSCheckout (cmsSessionId, cmsObjectId, rootWithDescendants);
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>cmsSessionId</code>	<code>int</code>	No	The ID of the CMS Session
<code>cmsObjectId</code>	<code>int</code>	No	The ID of the CMS Object
<code>rootWithDescendants</code>	<code>bool</code>	No	True if checked out root with descendants

## CMSCheckin

### Description

Checks in a file into the CMS

### Returns

Returns `FE_Success` if the operation is successful, else sets `FA_errno` to `Constants.FE_CMSFailedCheckin`

### Syntax

```
CMSCheckin (cmsSessionId, objectId, checkinParam);
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>cmsSessionId</code>	<code>int</code>	No	The ID of the CMS Session

Parameter name	Data Type	Optional	Description
cmsObjectId	int	No	The ID of the CMS Object
checkinParam	IdValue Pairs	No	Id value pairs to specify the checkin parameter. The valid Ids are: Constants.FV_CMSCCommandCheckinMakeCurrentVersionId Constants.FV_CMSCCommandCheckinKeepLocalCopyId Constants.FV_CMSCCommandCheckinMinorVersionId Constants.FV_CMSCCommandCheckinVersionLabelId Constants.FV_CMSCCommandCheckinDescriptionId Constants.FV_CMSCCommandCheckinCommentId

## CMSCancelCheckout

### Description

Cancels check out of a file from the CMS

### Returns

Returns FE\_Success if the operation is successful, else sets FA\_errno to Constants.FE\_CMSFailedCancelCheckout

### Syntax

```
CMSCancelCheckout (cmsSessionId, objectId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object

## CMSDelete

### Description

Deletes a file or a folder from CMS

### Returns

Returns FE\_Success if the operation is successful, else sets FA\_errno to Constants.FE\_CMSErrorDelete

### Syntax

```
CMSDelete (cmsSessionId, cmsObjectId, deleteParams);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
deleteParams	IdValue Pairs	No	Id value pairs to specify the delete parameter. The valid Ids are: Constants.FV_CMSErrorDeleteAllVersionId Constants.FV_CMSErrorDeleteAllDependentsId

## CMSOpenFile

### Description

Opens a file or a book from CMS in FrameMaker

### Returns

Returns the handle of the file or book if the operation is successful. Else sets FA\_errno to Constants.FE\_CMSErrorOpenFile

### Syntax

```
CMSOpenFile (cmsSessionId, cmsObjectId, openParams);
```

## Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
openParams	IdValue Pairs	No	Id value pairs to specify the open parameter. The valid Ids are: Constants.FV_CMSSCommandOpenReadOnlyId Constants.FV_CMSSCommandSilentOpenId

## CMSUploadObject

### Description

Uploads a file or a folder into the CMS

### Returns

Returns FE\_Success if the operation is successful. Else sets FA\_errno to Constants.FE\_CMSSFailedUploadObject

### Syntax

```
CMSUploadObject (cmsSessionId, cmsObjectId, localFilePath);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
localFilePath	string	No	The full pathname of the file or folder to upload

## CMSDownloadObject

### Description

Downloads a file from the CMS

**Returns**

Returns the local file path of the downloaded file if the operation is successful. Else sets FA\_errno to Constants.FE\_CMSFailedDownloadObject

**Syntax**

```
CMSDownloadObject (cmsSessionId, cmsObjectId);
```

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object

**GetCMSObjectFromPath****Description**

Gets CMS object from a URL path

**Returns**

Returns the handle of a CMS object if the operation is successful. Else sets FA\_errno to Constants.FE\_CMSFailedGetItemFrompath

**Syntax**

```
GetCMSObjectFromPath (cmsSessionId,urlPath);
```

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
urlPath	string	No	The url pathname of the file or folder

# Legal notices

For legal notices, see [Legal Notices](#)