

ADOBE® DREAMWEAVER® CS4

API Reference

© 2008 Adobe Systems Incorporated. All rights reserved.

Adobe® Dreamweaver® CS4 API 참조 설명서(Windows® 및 Mac OS용)

이 설명서가 일반 사용자 계약을 포함하는 소프트웨어와 함께 배포될 경우 이 설명서와 설명서에서 설명하는 소프트웨어는 허가 하에 제공되며 사용권 조항에 따라서만 사용하거나 복사할 수 있습니다. 사용권이 허가된 경우를 제외하고는 Adobe Systems Incorporated의 사전 서면 허가 없이 이 설명서의 어떠한 부분도 검색 시스템으로 재생하거나 저장할 수 없으며 전자, 기계, 녹음, 기타 어떤 형식이나 방법으로도 전달할 수 없습니다. 이 설명서가 최종 사용자 사용권 계약서가 포함된 소프트웨어와 함께 배포되지 않는 경우에도 설명서 내용은 저작권법의 보호를 받습니다. 이 설명서의 내용은 정보 사용 목적으로만 제공되며 사용자에게 통보 없이 변경될 수 있습니다. 또한 Adobe Systems Incorporated가 법적 책임은 지는 것으로 해석되어서는 안 됩니다. Adobe Systems Incorporated는 이 설명서에 있을 수 있는 정보의 오류나 부정확성에 대해 어떠한 책임과 의무도 지지 않습니다.

프로젝트에 포함하려는 기존 아트웍 또는 이미지는 저작권 법으로 보호받고 있으므로 주의하십시오. 허가 없이 새 작업에 이러한 자료를 포함시킬 경우 저작권자의 권리를 침해할 수 있습니다. 필요한 승인을 저작권 소유자로부터 받으시기 바랍니다. 샘플 템플릿에 언급된 회사 이름은 예를 들기 위해 사용되었으며 실제 조직과는 아무런 관련이 없습니다.

Adobe, the Adobe logo, ColdFusion, Dreamweaver, Fireworks, Flash, Photoshop, and Shockwave are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States.

Java is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries. Windows is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries. Macintosh and Mac OS are trademarks of Apple Inc., registered in the United States and other countries. All other trademarks are the property of their respective owners.

This work is licensed under the Creative Commons Attribution Non-Commercial 3.0 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/3.0/us/>

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

The Graphics Interchange Format © is the Copyright property of CompuServe Incorporated.

GIF is a Service Mark property of CompuServe Incorporated.

MPEG Layer-3 audio compression technology licensed by Fraunhofer IIS and Thomson Multimedia (<http://www.mp3licensing.com>). You cannot use the MP3 compressed audio within the Software for real time or live broadcasts. If you require an MP3 decoder for real time or live broadcasts, you are responsible for obtaining this MP3 technology license.

Speech compression and decompression technology licensed from Nellymoser, Inc. (www.nellymoser.com)

Video in Flash Player is powered by On2 TrueMotion video technology. © 1992-2005 On2 Technologies, Inc. All Rights Reserved. <http://www.on2.com>.

This product includes software developed by the OpenSymphony Group (<http://www.opensymphony.com/>)

 Sorenson Spark

Sorenson Spark™ video compression and decompression technology licensed from Sorenson Media, Inc.

The Proximity/Merriam-Webster Inc./Franklin Electronic Publishers Inc. Database © 1990/1994 Merriam-Webster Inc./Franklin Electronic Publishers Inc., © 1994. All Rights Reserved. Proximity Technology Inc. The Proximity/Merriam-Webster Inc./Franklin Electronic Publishers Inc. © 1990 Williams Collins Sons & Co. Ltd. © 1997 - All rights reserved Proximity Technology Inc. © 1990 Williams Collins Sons & Co. Ltd. © 1990 - All rights reserved Proximity Technology Inc. © Oxford University Press © 2000. All rights reserved Proximity Technology Inc. © 1990 IDE a.s. © 1990 - All rights reserved Proximity Technology Inc.

This product includes software developed by Fourthought, Inc. (<http://www.fourthought.com>).

This product includes software developed by CollabNet (<http://www.Collab.Net/>).

Adobe Systems Incorporated, 345 Park Avenue, San Jose, California 95110, USA.

Notice to U.S. Government End Users. The Software and Documentation are "Commercial Items," as that term is defined at 48 C.F.R. §2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R.

§227.7202, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the terms and conditions herein. Unpublished rights reserved under the copyright laws of the United States. Adobe Systems Incorporated, 345 Park Avenue, San Jose, CA 95110-2704, USA. For U.S. Government End Users, Adobe agrees to comply with all applicable equal opportunity laws including, if appropriate, the provisions of Executive Order 11246, as amended, Section 402 of the Vietnam Era Veterans Readjustment Assistance Act of 1974 (38 USC 4212), and Section 503 of the Rehabilitation Act of 1973, as amended, and the regulations at 41 CFR Parts 60-1 through 60-60, 60-250, and 60-741. The affirmative action clause and regulations contained in the preceding sentence shall be incorporated by reference.

목차

1장: 소개

Extension 정보	1
Dreamweaver 확장	1
Extension 개발자를 위한 추가 리소스	1
Dreamweaver CS4에 새로 추가된 함수	2
제거된 함수	4
이 설명서에서 사용된 규칙	5

2장: 파일 I/O API

구성 폴더 정보	7
파일 I/O API 정보	7

3장: HTTP API

HTTP API 작동 방식	16
HTTP API	16

4장: 디자인 노트 API

디자인 노트 작동 방식	22
디자인 노트 JavaScript API	22
디자인 노트 C API	27

5장: Fireworks 통합

FWLaunch API	33
--------------------	----

6장: Flash 통합

Flash 객체 API	39
Flash 패널 및 대화 상자 함수	41

7장: Photoshop 통합

스마트 오브젝트가 동작하는 방법	49
스마트 오브젝트 API	49

8장: 데이터베이스 API

데이터베이스 API 함수 작동 방식	53
데이터베이스 연결 함수	53
데이터베이스 액세스 함수	65

9장: 데이터베이스 연결 API

새 연결 유형 선택	76
새 연결 유형 개발	76
연결 API	77

생성된 포함 파일	80
연결 유형에 대한 정의 파일	81
 10장: 소스 제어 통합 API	
Dreamweaver의 소스 제어 통합 작동 방법	83
소스 제어 시스템 기능 추가	83
소스 제어 통합 API 필수 함수	84
소스 제어 통합 API 선택적 함수	89
활성자	96
 11장: 응용 프로그램	
외부 응용 프로그램 함수	102
전역 응용 프로그램 함수	110
Bridge 통신 함수	115
 12장: 작업 영역	
작업 내역 함수	117
객체 삽입 함수	125
키보드 함수	127
메뉴 함수	134
결과 윈도우 함수	135
전환 함수	147
툴바 함수	165
윈도우 함수	171
정보 막대 함수	181
관련 파일 함수	182
수직 분할 뷰 함수	183
코드 축소 함수	185
코드 뷰 툴바 함수	191
색상 함수	195
 13장: 사이트	
보고 함수	197
사이트 함수	198
 14장: 문서	
변환 함수	229
명령 함수	230
파일 조작 함수	231
전역 문서 함수	245
경로 함수	252
선택 함수	255
문자열 조작 함수	260

변환 함수	263
XSLT 함수	265
15장: 페이지 내용	
에셋 패널 함수	268
비헤이비어 함수	275
클립보드 함수	283
라이브러리 및 템플릿 함수	287
코드 단편 패널 함수	292
Spry 위젯 편집 함수	296
Spry 위젯 삽입 함수	298
브라우저 호환성 확인 함수	300
16장: 동적 문서	
서버 구성 요소 함수	308
데이터 소스 함수	309
Extension Data Manager 함수	310
라이브 데이터 함수	312
라이브 뷰 함수	316
서버 비헤이비어 함수	322
서버 모델 함수	323
17장: 디자인	
CSS 레이아웃 함수	329
프레임 및 프레임세트 함수	347
레이어 및 이미지 맵 함수	349
레이아웃 환경 함수	352
레이아웃 뷰 함수	357
확대/축소 함수	363
안내선 함수 및 속성	366
표 편집 함수	372
18장: 코드	
코드 함수	382
찾기 및 바꾸기 함수	386
일반 편집 함수	391
인쇄 함수	406
퀵 태그 편집기 함수	407
코드 뷰 함수	409
태그 편집기 및 태그 라이브러리 함수	426

19장: 활성화자

활성자 함수	431
--------------	-----

색인	467
----------	-----

1장: 소개

Adobe Dreamweaver CS4 API 참조 설명서에서는 API(응용 프로그램 프로그래밍 인터페이스)에 대해 설명합니다. API를 사용하면 Adobe® Dreamweaver® CS4 Extension을 개발하고 Dreamweaver 웹 페이지에 프로그램 코드를 추가할 때 다양한 지원 작업을 수행할 수 있습니다. 이러한 API에는 대부분의 Dreamweaver 핵심 기능에 액세스하는 데 사용할 수 있는 기본 JavaScript API가 포함됩니다. Dreamweaver의 핵심 기능은 일반적으로 메뉴 등을 통해 수행할 수 있는 모든 작업을 의미합니다. 또한 이러한 API에는 파일을 읽거나 쓰고, HTTP를 통해 정보를 전송하고, Fireworks 및 Flash와 통신하는 등의 일반적인 작업을 수행하는 데 사용할 수 있는 다양한 유틸리티 API도 포함됩니다.

광범위한 JavaScript API를 사용하면 다양한 일련의 간단한 작업을 수행할 수 있습니다. 사용자는 Dreamweaver 문서를 만들거나 편집할 때 이러한 작업을 수행합니다. 이러한 API 함수는 영향을 미치는 Dreamweaver 사용자 인터페이스 부분에 따라 그룹화됩니다. 예를 들어, JavaScript API에는 작업 영역 함수, 문서 함수, 디자인 함수 등이 포함됩니다. 이러한 API 함수를 사용하여 다음 작업을 비롯한 다양한 작업을 수행할 수 있습니다.

- 새 문서 열기
- 글꼴 크기 가져오기 또는 설정
- HTML 코드에서 검색 문자열 찾기
- 톨바 표시

Extension 정보

이 설명서에서는 사용자가 Dreamweaver, HTML, XML, JavaScript 프로그래밍 및 C 프로그래밍에 익숙하다고 가정합니다. 웹 응용 프로그램 작성을 위한 Extension을 작성하려면 하나 이상의 플랫폼에서 ASP(Active Server Pages), ASP.NET, PHP: Hypertext Preprocessor(PHP), ColdFusion, JSP(Java Server Pages) 등의 서버측 스크립팅에도 익숙해야 합니다.

Dreamweaver 확장

Dreamweaver Extension을 작성할 수 있는 Dreamweaver 프레임워크 및 API에 대한 자세한 내용은 **Dreamweaver** 확장을 참조하십시오. **Dreamweaver** 확장에서는 Dreamweaver의 다양한 기능을 구성하는 객체, 메뉴, 부동 패널, 서버 비헤이비어 등을 구현하기 위해 Dreamweaver에서 호출하는 API 함수에 대해 설명합니다. 이러한 API를 사용하여 객체, 메뉴, 부동 패널 또는 기타 기능을 제품에 추가할 수 있습니다. **Dreamweaver** 확장에서는 메뉴 항목 또는 문서 형식 등을 추가하기 위해 다양한 HTML 및 XML 파일에서 태그를 편집 및 추가하여 Dreamweaver를 사용자 정의하는 방법에 대해서도 설명합니다.

Extension 개발자를 위한 추가 리소스

Extension 작성에 참여하는 다른 개발자와 의견을 나누려면 Dreamweaver Extensibility 뉴스 그룹에 가입해야 합니다. <http://www.adobe.com/cfusion/webforums/forum/categories.cfm?forumid=12&catid=190>에서 이 뉴스 그룹의 웹 사이트에 액세스할 수 있습니다.

Dreamweaver CS4에 새로 추가된 함수

다음과 같은 새 함수가 Dreamweaver CS4 JavaScript API에 추가되었습니다. 각 머리글은 새로운 함수가 들어 있는 장과 단원을 나타냅니다.

동적 문서

다음 함수가 동적 문서 장에 추가되었습니다.

라이브 뷰 함수

- 316페이지의 “[dom.getDesignViewMode\(\)](#)”
- 317페이지의 “[dom.setDesignViewMode\(\)](#)”
- 317페이지의 “[dom.getLiveViewUsingServer\(\)](#)”
- 317페이지의 “[dom.setLiveViewUsingServer\(\)](#)”
- 318페이지의 “[dom.getLiveViewDefaultsToUsingServer\(\)](#)”
- 318페이지의 “[dom.getLiveViewDependentsUsingServer\(\)](#)”
- 318페이지의 “[dom.setLiveViewDependentsUsingServer\(\)](#)”
- 319페이지의 “[dom.showLiveViewParametersDialog\(\)](#)”
- 319페이지의 “[dom.browser.getSelection\(\)](#)”
- 319페이지의 “[dom.browser.getStatusText\(\)](#)”
- 320페이지의 “[dom.browser.getWindow\(\)](#)”
- 320페이지의 “[browserEle.loadHTML\(\)](#)”

Photoshop 통합

다음과 같은 새 함수가 Photoshop 통합 장에 추가되었습니다.

스마트 오브젝트 API

- 49페이지의 “[dreamweaver.assetPalette.canUpdateSmartObjectFromOriginal\(\)](#)”
- 49페이지의 “[dreamweaver.assetPalette.updateSmartObjectFromOriginal\(\)](#)”
- 50페이지의 “[dreamweaver.getSmartObjectState\(\)](#)”
- 50페이지의 “[dreamweaver.getSmartObjectOriginalWidth\(\)](#)”
- 51페이지의 “[dreamweaver.getImageWidth\(\)](#)”
- 51페이지의 “[dreamweaver.getImageHeight\(\)](#)”
- 51페이지의 “[dreamweaver.resolveOriginalAssetFileURLToAbsoluteLocalFilePath\(\)](#)”
- 52페이지의 “[dreamweaver.canUpdateSmartObjectFromOriginal\(\)](#)”
- 52페이지의 “[dreamweaver.updateSmartObjectFromOriginal\(\)](#)”

작업 영역

다음과 같은 새 함수가 작업 영역 장에 추가되었습니다.

관련 파일 함수

- 182페이지의 “[dreamweaver.getRelatedFiles\(\)](#)”
- 183페이지의 “[dreamweaver.openRelatedFile\(\)](#)”
- 183페이지의 “[dreamweaver.getActiveRelatedFilePath\(\)](#)”

정보 막대 함수

- 181페이지의 “[dom.showInfoBar\(\)](#)”
- 182페이지의 “[dom.hideInfoBar\(\)](#)”

수직 분할 뷰 함수

- 183페이지의 “[dreamweaver.getSplitViewOrientation\(\)](#)”
- 184페이지의 “[dreamweaver.setSplitViewOrientation\(\)](#)”
- 184페이지의 “[dreamweaver.getPrimaryView\(\)](#)”
- 184페이지의 “[dreamweaver.setPrimaryView\(\)](#)”
- 185페이지의 “[dom.isRelatedFileViewOpen\(\)](#)”

툴바 함수

- 170페이지의 “[dreamweaver.reloadToolbars\(\)](#)”

Flash 통합

다음과 같은 새 함수가 Flash 통합 장에 추가되었습니다.

Flash 패널 및 대화 상자 함수

- 41페이지의 “[dreamweaver.flash.newControl\(\)](#)”
- 43페이지의 “[dreamweaver.flash.requestStateChange\(\)](#)”
- 44페이지의 “[dreamweaver.flash.controlEvent\(\)](#)”
- 45페이지의 “[dreamweaver.flash.setMenu\(\)](#)”
- 47페이지의 “[dreamweaver.flash.evalScript\(\)](#)”
- 47페이지의 “[dreamweaver.flash.executeScript\(\)](#)”
- 48페이지의 “[dreamweaver.flash.controlExists](#)”

사이트

다음과 같은 새 함수가 사이트 장에 추가되었습니다.

- 215페이지의 “[site.getLocalRootURL\(\)](#)”
- 217페이지의 “[site.getSiteRootForURL\(\)](#)”

문서

다음과 같은 새 함수가 문서 장에 추가되었습니다.

- 256페이지의 “[dom.getSelectorsDefinedInStylesheet\(\)](#)”
- 252페이지의 “[dreamweaver.absoluteURLToDocRelative\(\)](#)”

코드

다음과 같은 새 함수가 코드 창에 추가되었습니다.

- 414페이지의 “[dom.source.doCodeNavItem\(\)](#)”

제거된 함수

다음 함수는 관련 기능이 제품에서 제거됨에 따라 Dreamweaver CS4 API에서 제거되었습니다.

site.setShowPageTitles()	site.removeLink()
site.canShowPageTitles()	site.invertSelection()
site.getShowPageTitles()	site.selectHomePage()
site.setShowHiddenFiles()	site.newHomePage()
site.getShowHiddenFiles()	site.setLayout()
site.setShowDependents()	dom.createLayoutCell()
site.getShowDependents()	dom.createLayoutTable()
site.canFindLinkSource()	dom.setColumnAutostretch()
site.findLinkSource()	dom.isColumnAutostretch()
site.viewAsRoot()	dom.doesColumnHaveSpacer()
site.setAsHomePage()	dom.doesGroupHaveSpacer()
site.canSetLayout()	dom.addSpacerToColumn()
site.canRemoveLink()	dom.removeSpacerFromColumn()
site.canChangeLink()	dom.removeAllSpacers()
site.canAddLink()	dom.makeCellWidthsConsistent()
site.addLinkToNewFile()	dom.insertFlashElement()
site.getLinkVisibility()	dreamweaver.exportCSS()(CS3에서 사용되지 않음)
site.setLinkVisibility()	dreamweaver.canExportCSS()(CS3에서 사용되지 않음)
site.saveAsImage()	dreamweaver.libraryPalette.deleteSelectedItem() (Dreamweaver 4에서 사용되지 않음)
dreamweaver.libraryPalette.getSelectedItem() (Dreamweaver 4에서 사용되지 않음)	dreamweaver.libraryPalette.newFromDocument() (Dreamweaver 4에서 사용되지 않음)
dreamweaver.libraryPalette.createFromDocument() (Dreamweaver 4에서 사용되지 않음)	dreamweaver.libraryPalette.renameSelectedItem() (Dreamweaver 4에서 사용되지 않음)

dreamweaver.templatePalette.deleteSelectedTemplate()(Dreamweaver 4에서 사용되지 않음)	dreamweaver.templatePalette.getSelectedTemplate()(Dreamweaver 4에서 사용되지 않음)
dreamweaver.templatePalette.renameSelectedTemplate()(Dreamweaver 4에서 사용되지 않음)	dreamweaver.getBehaviorEvent()(Dreamweaver 2에서 사용되지 않음)
dom.clipPasteText()(Dreamweaver 8에서 사용되지 않음)	dreamweaver.popupCommand()(Dreamweaver 3에서 사용되지 않음)
dom.serverModel.getServerLanguage()(Dreamweaver MX에서 사용되지 않음)	dom.serverModel.getServerExtension()(Dreamweaver MX에서 사용되지 않음)
dreamweaver.getObjectRefs()(Dreamweaver 3에서 사용되지 않음)	dreamweaver.getObjectTags()(Dreamweaver 3에서 사용되지 않음)
dreamweaver.getSelection()(Dreamweaver 3에서 사용되지 않음)	dreamweaver.nodeToOffsets()(Dreamweaver 3에서 사용되지 않음)
dreamweaver.offsetsToNode()(Dreamweaver 3에서 사용되지 않음)	dreamweaver.setSelection()(Dreamweaver 3에서 사용되지 않음)
dreamweaver.exportEditableRegionsAsXML()(Dreamweaver MX에서 사용되지 않음)	dreamweaver.cssStylePalette.getSelectedTarget()(Dreamweaver MX에서 사용되지 않음)
MMDB.getDriverUrlTemplateList()(Dreamweaver MX에서 사용되지 않음)	

이러한 사용되지 않는 API에 대한 자세한 내용은 [CS3용 Dreamweaver API 참조 설명서](#)를 참조하십시오.

이 설명서에서 사용된 규칙

인쇄 규칙

이 설명서에는 다음과 같은 인쇄 규칙이 사용되었습니다.

- 코드 글꼴은 코드 부분과 API 리터럴을 나타냅니다. 여기에는 클래스 이름, 메서드 이름, 함수 이름, 유형 이름, 스크립트, SQL 문, HTML과 XML의 태그 및 속성 이름 등이 포함됩니다.
- 기울임체 코드 글꼴은 코드에서 대체 가능한 항목을 나타냅니다.
- 연속 기호(↵)는 긴 코드 행이 둘 이상의 행으로 분리되었음을 나타냅니다. 이 설명서에서는 여백 제한으로 인해 원래는 연속되어야 하는 코드 행이 나뉘어져 표시됩니다. 코드 행을 복사하는 경우에는 연속 기호를 제거한 후 하나의 행으로 입력하십시오.
- 함수의 인수가 중괄호({ })로 묶여 있으면 해당 인수가 선택 사항임을 나타냅니다.

- 접두어 `dreamweaver.funcname`이 있는 함수 이름은 코드를 작성할 때 `dw.funcname`으로 줄여 쓸 수 있습니다. 이 설명서에서는 함수를 정의할 때나 색인에서 완전한 `dreamweaver` 접두어를 사용합니다. 그러나 많은 예제에서는 `dw` 접두어를 사용합니다.

명명 규칙

이 설명서에는 다음과 같은 명명 규칙이 사용됩니다.

- 개발자—`Extension` 작성을 담당하는 개발자
- 사용자—`Dreamweaver`를 사용하는 사람

2장: 파일 I/O API

Adobe® Dreamweaver® CS4에는 DWfile이라는 C 공유 라이브러리가 포함되어 있습니다. 객체, 명령, 비헤이비어, 데이터 변환기, 부동 패널 및 속성 관리자의 제작자는 DWfile을 통해 로컬 파일 시스템에서 파일을 읽고 작성할 수 있습니다. 이 장에서는 파일 I/O API와 그 사용 방법에 대해 설명합니다.

Dreamweaver에서 C 라이브러리와 JavaScript 인터프리터 사이의 상호 작용에 대한 전반적 내용은 **Dreamweaver** 확장의 "C 레벨 확장성"을 참조하십시오.

구성 폴더 정보

Microsoft Windows 2000과 Windows XP 및 Mac OS X 플랫폼에서는 사용자별로 구성 파일이 생성됩니다. Dreamweaver에서 구성 파일에 기록할 때는 항상 해당 사용자의 **Configuration** 폴더에 기록합니다. 마찬가지로 Dreamweaver에서 구성 파일을 읽는 경우 먼저 사용자의 **Configuration** 폴더에서 구성 파일을 찾은 다음 Dreamweaver Configuration 폴더에서 찾습니다. DWfile 함수도 동일한 방식을 사용합니다. 즉, Extension에서 Dreamweaver Configuration 폴더의 파일을 읽거나 기록하는 경우 사용자의 Configuration 폴더에도 액세스합니다. 다중 사용자 플랫폼의 구성 폴더에 대한 자세한 내용은 **Dreamweaver** 확장을 참조하십시오.

파일 I/O API 정보

파일 I/O API의 함수는 모두 DWfile 객체의 메서드입니다.

DWfile.copy()

지원 버전

Dreamweaver 3

설명

이 함수는 지정한 파일을 새 위치에 복사합니다.

인수

originalURL, copyURL

- **originalURL** 인수는 복사할 파일이며 file:// URL로 표시됩니다.
- **copyURL** 인수는 복사된 파일을 저장할 위치이며 file:// URL로 표시됩니다.

반환값

부울 값을 반환합니다. 복사가 성공하면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

다음 코드에서는 myconfig.cfg라는 파일을 myconfig_backup.cfg에 복사합니다.

```
var fileURL = "file:///c:/Config/myconfig.cfg";
var newURL = "file:///c:/Config/myconfig_backup.cfg";
DWfile.copy(fileURL, newURL);
```

DWfile.createFolder()

지원 버전

Dreamweaver 2

설명

이 함수는 지정한 위치에 폴더를 만듭니다.

인수

folderURL

- **folderURL** 인수는 폴더를 만들 위치이며, file:// URL로 표시됩니다.

반환값

부울 값을 반환합니다. 폴더가 성공적으로 생성되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

다음 코드에서는 C 드라이브의 최상위 위치에 tempFolder 폴더를 만들고 작업 성공 여부를 나타내는 경고 상자를 표시합니다.

```
var folderURL = "file:///c:/tempFolder";
if (DWfile.createFolder(folderURL)){
    alert("Created " + folderURL);
}else{
    alert("Unable to create " + folderURL);
}
```

DWfile.exists()

지원 버전

Dreamweaver 2

설명

이 함수는 지정한 파일이 있는지 테스트합니다.

인수

fileURL

- **fileURL** 인수는 요청한 파일이며 file:// URL로 표시됩니다.

반환값

부울 값을 반환합니다. 파일이 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

다음 코드에서는 mydata.txt 파일을 확인한 후 파일이 있는지 여부를 알려주는 경고 메시지를 표시합니다.

```
var fileURL = "file:///c:/temp/mydata.txt";
if (DWfile.exists(fileURL)){
    alert(fileURL + " exists!");
}else{
    alert(fileURL + " does not exist.");
}
```

DWfile.getAttributes()

지원 버전

Dreamweaver 2

설명

이 함수는 지정한 파일이나 폴더의 속성을 가져옵니다.

인수

fileURL

- **fileURL** 인수는 속성을 가져올 파일이나 폴더이며, file:// URL로 표시됩니다.

반환값

지정한 파일이나 폴더의 속성을 나타내는 문자열을 반환합니다. 해당 파일이나 폴더가 없으면 이 함수는 null 값을 반환합니다. 이 문자열에 다음과 같은 속성을 나타내는 문자를 사용할 수 있습니다.

- R은 읽기 전용입니다.
- D는 폴더입니다.
- H는 숨겨진 항목입니다.
- S는 시스템 파일 또는 폴더입니다.

예제

다음 코드에서는 mydata.txt 파일의 속성을 가져와서 파일이 읽기 전용이면 경고 상자를 표시합니다.

```
var fileURL = "file:///c:/temp/mydata.txt";  
var str = DWfile.getAttributes(fileURL);  
if (str && (str.indexOf("R") != -1)){  
    alert(fileURL + " is read only!");  
}
```

DWfile.getModificationDate()

지원 버전

Dreamweaver 2

설명

이 함수는 파일이 마지막으로 수정된 시간을 가져옵니다.

인수

fileURL

- **fileURL** 인수는 최종 수정 시간을 확인할 파일이며, file:// URL로 표시됩니다.

반환값

특정 기본 시간 후 경과된 시간 단위 수를 나타내는 16진수 숫자가 포함된 문자열을 반환합니다. 시간 단위와 기본 시간의 정확한 의미는 플랫폼에 따라 다릅니다. 예를 들어, Windows에서의 시간 단위는 100ns이며 기본 시간은 1600년 1월 1일입니다.

예제

이 함수에서 반환하는 값은 플랫폼에 따라 다르며 인식 가능한 날짜와 시간이 아니기 때문에 함수를 두 번 호출하여 반환값을 비교하는 것이 유용합니다. 다음 코드에서는 file1.txt와 file2.txt의 수정 날짜를 가져온 다음 최신 파일을 알려주는 경고 메시지를 표시합니다.

```
var file1 = "file:///c:/temp/file1.txt";
var file2 = "file:///c:/temp/file2.txt";
var time1 = DWfile.getModificationDate(file1);
var time2 = DWfile.getModificationDate(file2);
if (time1 == time2){
    alert("file1 and file2 were saved at the same time");
}else if (time1 < time2){
    alert("file1 older than file2");
}else{
    alert("file1 is newer than file2");
}
```

DWfile.getCreationDate()

지원 버전

Dreamweaver 4

설명

이 함수는 파일이 생성된 시간을 가져옵니다.

인수

fileURL

- fileURL 인수는 생성 시간을 확인할 파일이며, file:// URL로 표시됩니다.

반환값

특정 기본 시간 후 경과된 시간 단위 수를 나타내는 16진수 숫자가 포함된 문자열을 반환합니다. 시간 단위와 기본 시간의 정확한 의미는 플랫폼에 따라 다릅니다. 예를 들어, Windows에서의 시간 단위는 100ns이며 기본 시간은 1600년 1월 1일입니다.

예제

파일에 대해 이 함수와 DWfile.getModificationDate() 함수를 호출하여 생성 날짜와 수정 날짜를 비교할 수 있습니다.

```
var file1 = "file:///c:/temp/file1.txt";
var time1 = DWfile.getCreationDate(file1);
var time2 = DWfile.getModificationDate(file1);
if (time1 == time2){
    alert("file1 has not been modified since it was created");
}else if (time1 < time2){
    alert("file1 was last modified on " + time2);
}
```

DWfile.getCreationDateObj()

지원 버전

Dreamweaver MX

설명

이 함수는 파일이 생성된 시간을 나타내는 JavaScript 객체를 가져옵니다.

인수

fileURL

- fileURL 인수는 생성 시간을 확인할 파일이며, file:// URL로 표시됩니다.

반환값

지정한 파일이 생성된 날짜 및 시간을 나타내는 JavaScript Date 객체를 반환합니다.

DWfile.getModificationDateObj()

지원 버전

Dreamweaver MX

설명

이 함수는 파일이 마지막으로 수정된 시간을 나타내는 JavaScript Date 객체를 가져옵니다.

인수

fileURL

- fileURL 인수는 가장 최근에 수정된 시간을 확인할 파일이며, file:// URL로 표시됩니다.

반환값

지정한 파일이 마지막으로 수정된 날짜 및 시간을 나타내는 JavaScript Date 객체를 반환합니다.

DWfile.getSize()

지원 버전

Dreamweaver MX

설명

이 함수는 지정한 파일의 크기를 가져옵니다.

인수

fileURL

- fileURL 인수는 크기를 확인할 파일이며, file:// URL로 표시됩니다.

반환값

지정한 파일의 실제 크기를 바이트 단위로 나타내는 정수를 반환합니다.

DWfile.listFolder()

지원 버전

Dreamweaver 2

설명

이 함수는 지정한 폴더의 내용 목록을 가져옵니다.

인수

folderURL, {**constraint**}

- **folderURL** 인수는 내용 목록을 확인할 폴더이며, **file://** URL로 표시됩니다. 선택 사항으로 와일드카드 파일 마스크를 사용할 수 있습니다. 유효한 와일드 카드인 별표(*)는 하나 이상의 문자와 일치하고 물음표(?)는 단일 문자와 일치합니다.
- **constraint** 인수를 지정할 때는 "files"(파일만 반환) 또는 "directories"(폴더만 반환)로 지정해야 합니다. 이 인수를 생략하면 파일과 폴더가 모두 반환됩니다.

반환값

폴더 내용을 나타내는 문자열 배열을 반환합니다.

예제

다음 코드에서는 C:/temp 폴더에 있는 모든 텍스트(TXT) 파일의 목록을 가져오고 경고 메시지에 해당 목록을 표시합니다.

```
var folderURL = "file:///c:/temp";
var fileMask = "*.txt";
var list = DWfile.listFolder(folderURL + "/" + fileMask, "files");
if (list){
    alert(folderURL + " contains: " + list.join("\n"));
}
```

DWfile.read()

지원 버전

Dreamweaver 2

설명

이 함수는 지정한 파일의 내용을 문자열로 읽어옵니다.

인수

fileURL

- **fileURL** 인수는 읽을 파일이며, **file://** URL로 표시됩니다.

반환값

파일 내용이 포함된 문자열을 반환하거나, 읽기에 실패하는 경우에는 null 값을 반환합니다.

예제

다음 코드에서는 mydata.txt 파일을 읽고 읽기에 성공한 경우 경고 메시지에 파일 내용을 표시합니다.

```
var fileURL = "file:///c:/temp/mydata.txt";
var str = DWfile.read(fileURL);
if (str){
    alert(fileURL + " contains: " + str);
}
```

DWfile.remove()

지원 버전

Dreamweaver 3

설명

이 함수는 지정한 파일을 삭제합니다.

인수

fileURL

- **fileURL** 인수는 제거할 파일이며, file:// URL로 표시됩니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

다음 예제에서는 DWfile.getAttributes() 함수를 사용하여 파일이 읽기 전용인지 확인하고 confirm() 함수를 사용하여 [예/아니오] 대화 상자를 표시합니다.

```
function deleteFile(){
    var delAnyway = false;
    var selIndex = document.theForm.menu.selectedIndex;
    var selFile = document.theForm.menu.options[selIndex].value;
    if (DWfile.getAttributes(selFile).indexOf('R') != -1){
        delAnyway = confirm('This file is read-only. Delete anyway?');
        if (delAnyway){
            DWfile.remove(selFile);
        }
    }
}
```

DWfile.setAttributes()

지원 버전

Dreamweaver MX

설명

이 함수는 특정 파일의 시스템 레벨 속성을 설정합니다.

인수

fileURL, strAttrs

- **fileURL** 인수는 속성을 설정할 파일이며, file:// URL로 표시됩니다.
- **strAttrs** 인수는 **fileURL** 인수로 식별된 파일의 시스템 레벨 속성을 지정합니다. 다음 표에서는 유효한 속성 값과 그 의미에 대해 설명합니다.

속성 값	설명
R	읽기 전용
W	쓰기 가능(R 무시)
H	숨김
V	표시(H 무시)

strAttrs 문자열에 사용할 수 있는 값에는 R, W, H, V, RH, RV, WH 또는 WV가 있습니다.

서로 배타적인 R과 W는 함께 사용하지 마십시오. 두 값을 함께 사용하면 R이 무시되어 파일은 쓰기 가능(W)으로 설정됩니다. 마찬가지로 서로 배타적인 H와 V를 함께 사용하지 마십시오. 이 두 값을 함께 사용하면 H가 무시되어 파일은 표시(V)로 설정됩니다.

읽기/쓰기 속성인 R 또는 W를 지정하지 않은 상태로 H 또는 V를 지정하면 파일의 기존 읽기/쓰기 속성은 변경되지 않습니다. 마찬가지로 표시 여부 속성인 H 또는 V를 지정하지 않은 상태로 R 또는 W를 지정하면 해당 파일의 기존 표시 여부 속성은 변경되지 않습니다.

반환값

없음

DWfile.write()

지원 버전

Dreamweaver 2

설명

이 함수는 지정한 문자열을 지정한 파일에 기록합니다. 지정한 파일이 아직 존재하지 않으면 파일이 만들어집니다.

인수

fileURL, text, {mode}

- fileURL 인수는 데이터를 기록할 파일이며, file://URL로 표시됩니다.

참고: 경로에 공백이 포함되어 있는 경우 이 함수는 파일에 기록하지 않습니다.

- text 인수는 함수가 기록해야 하는 문자열을 나타냅니다.
- mode 인수를 지정할 때는 반드시 append로 지정해야 합니다. 이 인수를 생략하면 문자열이 파일 내용을 덮어씁니다.

반환값

부울 값을 반환합니다. 문자열이 파일에 성공적으로 기록되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

다음 코드에서는 문자열 xxx를 mydata.txt 파일에 기록한 다음 쓰기 작업이 성공한 경우 경고 메시지를 표시합니다. 그런 다음 문자열 aaa를 파일에 추가하고 쓰기 작업이 성공한 경우 두 번째 경고를 표시합니다. 이 스크립트를 실행하고 나면 mydata.txt 파일에는 xxxaaa 텍스트만 들어 있게 됩니다.

```
var fileURL = "file:///c:/temp/mydata.txt";  
if (DWfile.write(fileURL, "xxx")){  
    alert("Wrote xxx to " + fileURL);  
}  
if (DWfile.write(fileURL, "aaa", "append")){  
    alert("Appended aaa to " + fileURL);  
}
```

3장: HTTP API

확장 기능은 로컬 파일 시스템 내에서만 작동하도록 제한되지 않습니다. Adobe® Dreamweaver®에서는 HTTP(Hypertext Transfer Protocol)를 사용하여 웹 서버와 정보를 주고 받는 메커니즘을 제공합니다. 이 장에서는 HTTP API와 그 사용 방법에 대해 설명합니다.

HTTP API 작동 방식

HTTP API의 모든 함수는 MMHttp 객체의 메서드입니다. 이러한 함수는 대부분 URL을 인수로 사용하고 객체를 반환합니다. URL 인수의 기본 포트는 80입니다. 80 이외의 포트를 지정하려면 다음 예제에 표시된 대로 URL에 콜론과 포트 번호를 추가하십시오.

```
MMHttp.getText("http://www.myserver.com:8025");
```

객체를 반환하는 함수의 경우 객체에 statusCode 및 data 속성이 있습니다.

statusCode 속성은 작업의 상태를 나타냅니다. 속성 값으로 다음과 같은 값을 사용할 수 있지만 이 값으로 제한되지는 않습니다.

- 200: 정상 상태
- 400: 이해할 수 없는 요청
- 404: 요청한 URL을 찾을 수 없습니다.
- 405: 서버에서 요청된 메서드를 지원하지 않습니다.
- 500: 알 수 없는 서버 오류
- 503: 서버 용량 초과

서버의 상태 코드에 대한 전체 목록을 보려면 인터넷 서비스 공급자나 시스템 관리자에게 문의하십시오.

data 속성의 값은 함수에 따라 다르며, 사용 가능한 값은 개별 함수 목록에 지정되어 있습니다.

객체를 반환하는 함수에도 콜백 버전이 있습니다. 콜백 함수는 웹 서버가 HTTP 요청을 처리하는 동안 다른 함수가 실행될 수 있도록 합니다. Dreamweaver에서 여러 가지 HTTP 요청을 하는 경우에 이 기능이 유용합니다. 함수의 콜백 버전은 첫 번째 인수로 지정된 함수로 ID와 반환값을 직접 전달합니다.

HTTP API

이 단원에서는 MMHttp 객체의 메서드에 해당하는 함수에 대해 설명합니다.

MMHttp.clearServerScriptsFolder()

지원 버전

Dreamweaver MX

설명

현재 사이트에 대한 루트 폴더 아래에서 _mmServerScripts 폴더와 해당 파일을 모두 삭제합니다. 이 폴더는 로컬이거나 원격일 수 있습니다. _mmServerScripts 폴더는 Configuration/Connections/Scripts/server-model/_mmDBScripts 폴더에 있습니다.

인수

serverScriptsfolder

- **serverScriptsfolder** 인수는 서버 스크립트를 검색하고 제거할 특정 폴더의 이름을 지정하는 문자열입니다. 이때 폴더 이름은 응용 프로그램 서버의 **Configuration** 폴더에 대한 상대 경로로 지정합니다.

반환값

서버의 응답을 나타내는 객체를 반환합니다. 이 객체의 **data** 속성은 삭제된 스크립트의 내용을 포함하는 문자열입니다. 오류가 발생하면 반환되는 객체의 **statusCode** 속성에서 오류가 보고됩니다.

예제

Configuration/Menu 폴더의 메뉴 명령 파일에 있는 다음 코드를 메뉴에서 호출하면 **_mmServerScripts** 폴더의 파일이 모두 제거됩니다.

```
<!-- MENU-LOCATION=NONE -->
<html>
<head>
<TITLE>Clear Server Scripts</TITLE>
<SCRIPT SRC="ClearServerScripts.js"></SCRIPT>
<SCRIPT LANGUAGE="javascript">
</SCRIPT>
<body onLoad="MMHttp.clearServerScriptsFolder()">
</body>
</html>
```

MMHttp.clearTemp()

설명

이 함수를 실행하면 Dreamweaver 응용 프로그램 폴더에 있는 **Configuration/Temp** 폴더의 파일이 모두 삭제됩니다.

인수

없음

반환값

없음

예제

다음 코드를 **Configuration/Shutdown** 폴더 내의 파일에 저장하면 사용자가 Dreamweaver를 종료할 때 **Configuration/Temp** 폴더의 파일이 모두 제거됩니다.

```
<html>
<head>
<title>Clean Up Temp Files on Shutdown</title>
</head>
<body onLoad="MMHttp.clearTemp()">
</body>
</html>
```

MMHttp.getFile()

설명

이 함수는 지정된 URL에서 파일을 가져와 Dreamweaver 응용 프로그램 폴더에 있는 Configuration/Temp 폴더에 저장합니다. Dreamweaver에서는 서버의 폴더 구조를 모방하는 하위 폴더가 자동으로 만들어집니다. 예를 들어, 지정된 파일이 www.dreamcentral.com/people/index.html에 있으면 index.html 파일이 www.dreamcentral.com 폴더의 People 폴더에 저장됩니다.

인수

URL, {prompt}, {saveURL}, {titleBarLabel}

- URL 인수는 웹 서버에 대한 절대 URL입니다. URL에서 http://가 생략된 경우에도 HTTP 프로토콜로 간주됩니다.
- 선택적인 **prompt** 인수는 파일을 저장하라는 메시지를 표시할지 여부를 지정하는 부울 값입니다. **saveURL**이 Configuration/Temp 폴더 외부에 있는 경우 **prompt** 값 false는 보안상의 이유로 무시됩니다.
- 선택적인 **saveURL** 인수는 파일이 저장될 사용자 하드 디스크에서의 위치이며, file:// URL로 표시됩니다. **prompt** 값이 true이거나 **saveURL**이 Configuration/Temp 폴더 외부에 있으면 사용자는 [저장] 대화 상자에서 **saveURL**을 무시할 수 있습니다.
- 선택적인 **titleBarLabel** 인수는 [저장] 대화 상자의 제목 막대에 표시될 레이블입니다.

반환값

서버의 응답을 나타내는 객체를 반환합니다. 이 객체의 data 속성은 파일이 저장된 위치를 포함하는 문자열이며, file:// URL로 표시됩니다. 일반적으로 객체의 statusCode 속성에는 서버에서 받은 상태 코드가 포함됩니다. 그러나 해당 파일이 로컬 드라이브에 저장되는 동안 디스크 오류가 발생하면 statusCode 속성에는 작업이 성공적이지 않은 경우 다음 오류 코드 중 하나를 나타내는 정수가 포함됩니다.

- 1: 지정되지 않은 오류입니다.
- 2: 파일을 찾을 수 없습니다.
- 3: 경로가 잘못되었습니다.
- 4: 열려 있는 파일 수 제한에 도달했습니다.
- 5: 액세스가 거부되었습니다.
- 6: 파일 핸들이 잘못되었습니다.
- 7: 현재의 작업 폴더를 제거할 수 없습니다.
- 8: 폴더 항목이 더 이상 없습니다.
- 9: 파일 포인터를 설정하는 동안 오류가 발생했습니다.
- 10: 하드웨어 오류입니다.
- 11: 공유 위반입니다.
- 12: 잠금 위반입니다.
- 13: 디스크가 꽉 찼습니다.
- 14: 파일의 끝에 도달했습니다.

예제

다음 코드에서는 HTML 파일을 가져오고 모든 파일을 Configuration/Temp 폴더에 저장한 다음 HTML 파일의 로컬 복사본을 브라우저에서 엽니다.

```
var httpReply = MMHttp.getFile("http://www.dreamcentral.com/people/profiles/scott.html",
false);
if (Boolean == 200){
    var saveLoc = httpReply.data;
    dw.browseDocument (saveLoc);
}
```

MMHttp.getFileCallback()

설명

이 함수는 지정된 URL에서 파일을 가져와 Dreamweaver 응용 프로그램 폴더 내의 Configuration/Temp 폴더에 저장한 다음 요청 ID 및 응답 결과로 지정된 함수를 호출합니다. 파일을 로컬로 저장하면 서버의 폴더 구조를 모방하는 하위 폴더가 자동으로 만들어집니다. 예를 들어, 지정된 파일이 www.dreamcentral.com/people/index.html에 있으면 www.dreamcentral.com 폴더의 People 폴더에 index.html 파일이 저장됩니다.

인수

callbackFunction, URL, {prompt}, {saveURL}, {titleBarLabel}

- **callbackFunction** 인수는 HTTP 요청이 완료되는 경우 호출할 JavaScript 함수의 이름입니다.
- **URL** 인수는 웹 서버에 대한 절대 URL입니다. URL에서 http://가 생략된 경우에도 HTTP 프로토콜로 간주됩니다.
- 선택적인 **prompt** 인수는 파일을 저장하라는 메시지를 표시할지 여부를 지정하는 부울 값입니다. **saveURL** 인수가 Configuration/Temp 폴더 외부에 있는 위치를 지정하는 경우 **prompt** 값 false는 보안상의 이유로 무시됩니다.
- 선택적인 **saveURL** 인수는 파일이 저장될 사용자 하드 디스크에서의 위치이며, file:// URL로 표시됩니다. **prompt** 값이 true 이거나 **saveURL**이 Configuration/Temp 폴더 외부에 있으면 사용자는 [저장] 대화 상자에서 **saveURL**을 무시할 수 있습니다.
- 선택적인 **titleBarLabel** 인수는 [저장] 대화 상자의 제목 막대에 표시될 레이블입니다.

반환값

서버의 응답을 나타내는 객체를 반환합니다. 이 객체의 data 속성은 파일이 저장된 위치를 포함하는 문자열이며, file:// URL로 표시됩니다. 일반적으로 객체의 statusCode 속성에는 서버에서 받은 상태 코드가 포함됩니다. 그러나 해당 파일이 로컬 드라이브에 저장되는 동안 디스크 오류가 발생하면 statusCode 속성에는 오류 코드를 나타내는 정수가 포함됩니다. 가능한 오류 코드 목록은 18페이지의 “MMHttp.getFile()”을 참조하십시오.

MMHttp.getText()

지원 버전

Dreamweaver UltraDev 4, Dreamweaver MX에서 기능이 향상됨

설명

지정된 URL에서 문서의 내용을 검색합니다.

인수

URL, {serverScriptsFolder}

- **URL** 인수는 웹 서버에 대한 절대 URL입니다. URL에서 http://가 생략된 경우에도 HTTP 프로토콜로 간주됩니다.
- **serverScriptsFolder** 인수는 서버 스크립트를 검색할 특정 폴더의 이름을 지정하는 선택적 문자열입니다. 폴더 이름은 응용 프로그램 서버의 Configuration 폴더에 대한 상대 경로로 지정합니다. Dreamweaver에서는 스크립트를 검색할 때 FTP,

WebDAV 또는 원격 파일 시스템 등의 적절한 전송 프로토콜을 사용합니다. Dreamweaver에서는 이러한 파일을 현재 사이트의 루트 폴더 아래에 있는 `_mmServerScripts` 하위 폴더에 복사합니다.

오류가 발생하면 반환되는 객체의 `statusCode` 속성에서 오류가 보고됩니다.

MMHttp.getTextCallback()

지원 버전

Dreamweaver UltraDev 4, Dreamweaver MX에서 기능이 향상됨

설명

지정된 URL에서 문서의 내용을 검색하여 지정된 함수에 전달합니다.

인수

`callbackFunc`, `URL`, `{serverScriptsFolder}`

- `callbackFunc` 인수는 HTTP 요청이 완료될 때 호출할 JavaScript 함수입니다.
- `URL` 인수는 웹 서버에 대한 절대 URL입니다. URL에서 `http://`가 생략된 경우에도 HTTP 프로토콜로 간주됩니다.
- `serverScriptsFolder` 인수는 서버 스크립트를 검색할 특정 폴더의 이름을 지정하는 선택적 문자열입니다. 폴더 이름은 응용 프로그램 서버의 `Configuration` 폴더에 대한 상대 경로로 지정합니다. Dreamweaver에서는 스크립트를 검색할 때 FTP, WebDAV 또는 원격 파일 시스템 등의 적절한 전송 프로토콜을 사용합니다. Dreamweaver에서는 이러한 파일을 검색하여 `callbackFunc`가 나타내는 함수에 전달합니다.

오류가 발생하면 반환되는 객체의 `statusCode` 속성에서 오류가 보고됩니다.

MMHttp.postText()

지원 버전

Dreamweaver UltraDev 4, Dreamweaver MX에서 기능이 향상됨

설명

HTTP Post를 수행하여 지정된 URL에 지정된 데이터를 게시합니다. 게시 작업과 연관된 데이터는 일반적으로 양식 인코딩 텍스트이지만 서버가 수신할 수 있는 모든 유형의 데이터일 수 있습니다.

인수

`URL`, `dataToPost`, `{contentType}`, `{serverScriptsFolder}`

- `URL` 인수는 웹 서버에 대한 절대 URL입니다. URL에서 `http://`가 생략된 경우에도 HTTP 프로토콜로 간주됩니다.
- `dataToPost` 인수는 게시할 데이터입니다. 세 번째 인수가 "application/x-www-form-urlencoded"이거나 생략되는 경우 `dataToPost`는 RFC 1866 사양(www.faqs.org/rfcs/rfc1866.html)의 8.2.1 섹션에 따라 양식으로 인코딩되어야 합니다.
- 선택적인 `contentType` 인수는 게시할 데이터의 내용 유형입니다. 이 인수를 생략하면 기본값은 "application/x-www-form-urlencoded"입니다.
- `serverScriptsFolder` 인수는 데이터를 게시할 특정 폴더의 이름을 지정하는 선택적 문자열입니다. 해당 폴더는 데이터를 게시할 응용 프로그램 서버의 `Configuration` 폴더에 대한 상대 경로로 지정합니다. Dreamweaver에서는 데이터를 게시하기 위해 적절한 전송 프로토콜(예: FTP, WebDAV 또는 원격 파일 시스템)을 사용합니다.

오류가 발생하면 반환되는 객체의 `statusCode` 속성에서 오류가 보고됩니다.

예제

다음의 MMHttp.postText() 함수 호출 예제에서는 개발자가 myScripts.cfm 파일을 로컬 컴퓨터의 Configuration 폴더 아래에 있는 DeployScripts 폴더에 저장했다고 가정합니다.

```
MMHttp.postText(
    "http://ultraqa8/DeployScripts/myScripts.cfm",
    "arg1=Foo",
    "application/x-www-form-urlencoded",
    "Configuration/DeployScripts/"
)
```

Dreamweaver에서 이 함수를 호출하면 다음과 같은 작업이 순서대로 수행됩니다.

- 1 로컬 컴퓨터의 Configuration/DeployScripts 폴더에 있는 myScripts.cfm 파일이 ultraqa8 웹 사이트의 루트 폴더에 있는 하위 폴더인 DeployScripts 폴더로 복사됩니다. Dreamweaver에서는 파일을 배포하기 위해 사이트 구성 속성에 지정된 프로토콜을 사용합니다.
- 2 Dreamweaver에서는 HTTP 프로토콜을 사용하여 웹 서버에 arg1=Foo 데이터를 게시합니다.
- 3 게시 요청의 결과로 ultraqa8에 대한 웹 서버에서 arg1 데이터를 사용하여 myScripts.cfm 스크립트를 실행합니다.

MMHttp.postTextCallback()**지원 버전**

Dreamweaver UltraDev 4, Dreamweaver MX에서 기능이 향상됨

설명

HTTP Post를 수행하여 지정된 URL에 텍스트를 게시하고 서버의 응답을 지정된 함수에 전달합니다. 게시 작업과 연관된 데이터는 일반적으로 양식 인코딩 텍스트이지만 서버가 수신할 수 있는 모든 유형의 데이터일 수 있습니다.

인수

callbackFunc, URL, dataToPost, {contentType}, {serverScriptsFolder}

- **callbackFunc** 인수는 HTTP 요청이 완료될 때 호출할 JavaScript 함수의 이름입니다.
- **URL** 인수는 웹 서버에 대한 절대 URL입니다. URL에서 http://가 생략된 경우에도 HTTP 프로토콜로 간주됩니다.
- **dataToPost** 인수는 게시할 데이터입니다. 세 번째 인수가 "application/x-www-form-urlencoded"이거나 생략되는 경우 data는 RFC 1866 사양(www.faqs.org/rfcs/rfc1866.html)의 8.2.1 섹션에 따라 양식으로 인코딩되어야 합니다.
- 선택적인 **contentType** 인수는 게시할 데이터의 내용 유형입니다. 이 인수를 생략하면 기본값은 "application/x-www-form-urlencoded"입니다.
- **serverScriptsFolder** 인수는 선택적 문자열입니다. 이 인수는 데이터를 게시할 특정 폴더의 이름을 지정합니다. 해당 폴더는 응용 프로그램 서버의 Configuration 폴더에 대한 상대 경로로 지정합니다. Dreamweaver에서는 데이터를 게시하기 위해 FTP, WebDAV, 원격 파일 시스템 등의 적절한 전송 프로토콜을 사용합니다. Dreamweaver에서는 이러한 데이터를 검색하여 callbackFunc가 나타내는 함수에 전달합니다.

오류가 발생하면 반환되는 객체의 statusCode 속성에서 오류가 보고됩니다.

4장: 디자인 노트 API

Adobe® Dreamweaver®, Adobe® Fireworks® 및 Adobe® Flash®는 문서에 대한 추가 정보를 저장하고 검색하는 방법을 웹 디자이너와 개발자에게 제공합니다. 이러한 정보는 디자인 노트라는 파일에 저장되며 검토용 주석, 변경 내용에 관한 기록, GIF나 JPEG의 소스 파일과 같은 문서에 대한 추가 정보를 제공합니다.

Dreamweaver의 디자인 노트 기능을 사용하는 방법에 대한 자세한 내용은 **Dreamweaver** 사용 설명서를 참조하십시오.

디자인 노트 작동 방식

각 디자인 노트 파일은 한 문서에 대한 정보를 저장합니다. 폴더에 있는 하나 이상의 문서가 디자인 노트 파일과 연관된 경우 디자인 노트 파일을 저장할 수 있는 `_notes` 하위 폴더가 만들어집니다. `_notes` 폴더와 이 폴더에 들어 있는 디자인 노트 파일은 [사이트] 패널에는 표시되지 않지만 Finder(Macintosh)나 Windows 탐색기에 나타납니다. 디자인 노트 파일 이름은 주 파일 이름에 `.mno` 확장명을 추가하여 지정합니다. 예를 들어, `avocado8.gif`와 연관된 디자인 노트 파일은 `avocado8.gif.mno`입니다.

디자인 노트 파일은 일련의 키/값 쌍으로 정보를 저장하는 XML 파일입니다. 키는 저장되는 정보의 유형을 설명하고 값은 정보 자체를 나타냅니다. 키는 64자로 제한됩니다.

다음 예제에서는 `foghorn.gif.mno`에 대한 디자인 노트 파일을 보여 줍니다.

```
<?xml version="1.0" encoding="iso-8859-1" ?>
<info>
  <infoitem key="FW_source" value="file:///C:/sites/dreamcentral/images/sourceFiles/-
    foghorn.png" />
  <infoitem key="Author" value="Heidi B." />
  <infoitem key="Status" value="Final draft, approved by Jay L." />
</info>
```

디자인 노트 JavaScript API

디자인 노트 JavaScript API의 모든 함수는 MMNotes 객체의 메서드입니다. MMNotes는 Extension 제작자가 디자인 노트 파일을 읽고 쓰는 데 사용할 수 있는 C 공유 라이브러리입니다. DWfile 공유 라이브러리처럼 MMNotes에는 함수를 호출할 수 있는 JavaScript API가 있습니다. 이러한 함수는 라이브러리의 객체, 명령, 비헤이비어, 부동 패널, 속성 관리자 및 데이터 변환기에서 호출됩니다. MMNotes 공유 라이브러리는 Dreamweaver가 설치되지 않은 경우에도 독립적으로 사용할 수 있습니다.

MMNotes.close()

설명

이 함수는 지정된 디자인 노트 파일을 닫고 변경 내용을 저장합니다. 모든 키/값 쌍이 제거되면 디자인 노트 파일이 삭제됩니다. 삭제된 파일이 `_notes` 폴더에 있는 마지막 디자인 노트 파일인 경우 해당 폴더도 삭제됩니다.

참고: 디자인 노트 작업이 완료되면 항상 MMNotes.close() 함수를 호출하여 Dreamweaver에서 파일에 쓰도록 해야 합니다.

인수

fileHandle

- **fileHandle** 인수는 MMNotes.open() 함수가 반환하는 파일 핸들입니다.

반환값

없음

예제

26페이지의 “[MMNotes.set\(\)](#)”을 참조하십시오.

MMNotes.filePathToLocalURL()

설명

이 함수는 지정된 로컬 드라이브 경로를 file:// URL로 변환합니다.

인수**drivePath**

- **drivePath** 인수는 전체 드라이브 경로를 포함하는 문자열입니다.

반환값

지정된 파일의 file:// URL을 포함하는 문자열을 반환합니다.

예제

MMNotes.filePathToLocalURL('C:\sites\webdev\index.htm')을 호출하면 "file:///c:/sites/webdev/index.htm"이 반환됩니다.

MMNotes.get()

설명

이 함수는 지정된 디자인 노트 파일에서 지정된 키 값을 가져옵니다.

인수**fileHandle, keyName**

- **fileHandle** 인수는 MMNotes.open()이 반환하는 파일 핸들입니다.
- **keyName** 인수는 키 이름을 포함하는 문자열입니다.

반환값

키 값을 포함하는 문자열을 반환합니다.

예제

24페이지의 “[MMNotes.getKeys\(\)](#)”를 참조하십시오.

MMNotes.getKeyCount()

설명

이 함수는 지정된 디자인 노트 파일에서 키/값 쌍의 수를 가져옵니다.

인수**fileHandle**

- **fileHandle** 인수는 MMNotes.open() 함수가 반환하는 파일 핸들입니다.

반환값

디자인 노트 파일에 있는 키/값 쌍의 수를 나타내는 정수를 반환합니다.

MMNotes.getKeys()

설명

이 함수는 디자인 노트 파일에 있는 모든 키의 목록을 가져옵니다.

인수**fileHandle**

- **fileHandle** 인수는 MMNotes.open() 함수가 반환하는 파일 핸들입니다.

반환값

각 문자열이 키 이름을 포함하는 문자열 배열을 반환합니다.

예제

사용자 정의 부동 패널에서 다음 코드를 사용하면 활성 문서에 대한 디자인 노트 정보를 표시할 수 있습니다.

```
var noteHandle = MMNotes.open(dw.getDocumentDOM().URL);
var theKeys = MMNotes.getKeys(noteHandle);
var noteString = "";
var theValue = "";
for (var i=0; i < theKeys.length; i++){
    theValue = MMNotes.get(noteHandle,theKeys[i]);
    noteString +=theKeys[i] + " = " + theValue + "\n";
}
document.theForm.bigTextField.value = noteString;
// always close noteHandle
MMNotes.close(noteHandle);
```

MMNotes.getSiteRootForFile()

설명

이 함수는 지정된 디자인 노트 파일의 사이트 루트를 확인합니다.

인수**fileURL**

- **fileURL** 인수는 로컬 파일의 경로이며, file:// URL로 표시됩니다.

반환값

file:// URL로 표시되는 사이트의 로컬 루트 폴더 경로를 포함하는 문자열을 반환하거나, Dreamweaver가 설치되어 있지 않거나 디자인 노트 파일이 Dreamweaver로 정의한 사이트 외부에 있는 경우에는 빈 문자열을 반환합니다. 이 함수는 Dreamweaver에 정의된 모든 사이트를 검색합니다.

MMNotes.getVersionName()

설명

이 함수는 MMNotes 공유 라이브러리를 구현한 응용 프로그램을 나타내는 MMNotes 공유 라이브러리 버전 이름을 가져옵니다.

인수

없음

반환값

MMNotes 공유 라이브러리를 구현한 응용 프로그램의 이름을 포함하는 문자열을 반환합니다.

예제

Dreamweaver 명령, 객체, 비헤이비어, 속성 관리자, 부동 패널 또는 데이터 변환기에서 MMNotes.getVersionName() 함수를 호출하면 "Dreamweaver"가 반환됩니다. Fireworks는 Dreamweaver 엔지니어링 팀에서 만든 동일한 버전의 라이브러리를 사용하기 때문에 Fireworks에서 MMNotes.getVersionName() 함수를 호출해도 "Dreamweaver"가 반환됩니다.

MMNotes.getVersionNum()

설명

이 함수는 MMNotes 공유 라이브러리의 버전 번호를 가져옵니다.

인수

없음

반환값

버전 번호를 포함하는 문자열을 반환합니다.

MMNotes.localURLToFilePath()

설명

이 함수는 지정된 file:// URL을 로컬 드라이브 경로로 변환합니다.

인수

fileURL

- fileURL 인수는 로컬 파일의 경로이며, file:// URL로 표시됩니다.

반환값

지정된 파일의 로컬 드라이브 경로를 포함하는 문자열을 반환합니다.

예제

MMNotes.localURLToFilePath("file:///MacintoshHD/images/moon.gif")를 호출하면 "MacintoshHD:images:moon.gif"가 반환됩니다.

MMNotes.open()

설명

이 함수는 지정된 파일과 연관된 디자인 노트 파일을 열거나, 해당 디자인 노트 파일이 없는 경우 새로 만듭니다.

인수

filePath, {bForceCreate}

- **filePath** 인수는 디자인 노트 파일이 연관된 주 파일의 경로이며, **file://** URL로 표시됩니다.
- **bForceCreate** 인수는 디자인 노트가 사이트에 대해 해제되어 있거나 **filePath** 인수가 사이트와 연관되어 있지 않은 경우에도 노트를 만들지 여부를 나타내는 부울 값입니다.

반환값

디자인 노트 파일의 파일 핸들을 반환하거나, 해당 파일이 열리지 않았거나 만들어지지 않은 경우에는 0을 반환합니다.

예제

26페이지의 “[MMNotes.set\(\)](#)”을 참조하십시오.

MMNotes.remove()

설명

이 함수는 지정된 디자인 노트 파일에서 지정된 키와 그 값을 제거합니다.

인수

fileHandle, **keyName**

- **fileHandle** 인수는 `MMNotes.open()` 함수가 반환하는 파일 핸들입니다.
- **keyName** 인수는 제거할 키 이름을 포함하는 문자열입니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

MMNotes.set()

설명

이 함수는 디자인 노트 파일에서 키/값 쌍 하나를 만들거나 업데이트합니다.

인수

fileHandle, **keyName**, **valueString**

- **fileHandle** 인수는 `MMNotes.open()` 함수가 반환하는 파일 핸들입니다.
- **keyName** 인수는 키 이름을 포함하는 문자열입니다.
- **valueString** 인수는 값을 포함하는 문자열입니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

예제

다음 예제에서는 peakhike99/index.html이라는 dreamcentral 사이트의 파일과 연관된 디자인 노트 파일을 열어 새로운 키/값 쌍을 추가하고 기존 키 값을 변경한 다음 디자인 노트 파일을 닫습니다.

```
var noteHandle = MMNotes.open('file:///c:/sites/dreamcentral/peakhike99/
    index.html',true);
if(noteHandle > 0){
    MMNotes.set(noteHandle,"Author","M. G. Miller");
    MMNotes.set(noteHandle,"Last Changed","August 28, 1999");
    MMNotes.close(noteHandle);
}
```

디자인 노트 C API

MMNotes 공유 라이브러리는 JavaScript API뿐만 아니라 다른 응용 프로그램이 디자인 노트 파일을 만드는 데 사용할 수 있는 C API도 제공합니다. Dreamweaver에서 MMNotes 공유 라이브러리를 사용하는 경우에는 JavaScript 버전의 함수가 C 함수를 호출하기 때문에 이러한 C 함수를 직접 호출할 필요가 없습니다.

이 단원에는 함수, 인수 및 반환값에 대한 설명이 포함되어 있습니다. 함수 및 데이터 형식에 대한 정의는 Dreamweaver 응용 프로그램 폴더에 있는 Extending/c_files 폴더의 MMInfo.h 파일에 들어 있습니다.

void CloseNotesFile()

설명

이 함수는 지정된 디자인 노트 파일을 닫고 변경 내용을 저장합니다. 디자인 노트 파일에서 모든 키/값 쌍이 제거되면 디자인 노트 파일이 삭제됩니다. 마지막 디자인 노트 파일이 삭제되면 _notes 폴더가 삭제됩니다.

인수

noteHandle

- **noteHandle** 인수는 OpenNotesFile() 함수가 반환하는 파일 핸들입니다.

반환값

없음

BOOL FilePathToLocalURL()

설명

이 함수는 지정된 로컬 드라이브 경로를 file:// URL로 변환합니다.

인수

const char* drivePath, char* localURLBuf, int localURLMaxLen

- **drivePath** 인수는 전체 드라이브 경로를 포함하는 문자열입니다.
- **localURLBuf** 인수는 file:// URL이 저장되는 버퍼입니다.
- **localURLMaxLen** 인수는 localURLBuf의 최대 크기입니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다. `localURLBuf` 인수는 `file:// URL` 값을 받습니다.

BOOL GetNote()

설명

이 함수는 지정된 디자인 노트 파일에서 지정된 키 값을 가져옵니다.

인수

`FileHandle noteHandle`, `const char keyName[64]`, `char* valueBuf`, `int valueBufLength`

- `noteHandle` 인수는 `OpenNotesFile()` 함수가 반환하는 파일 핸들입니다.
- `keyName[64]` 인수는 키 이름을 포함하는 문자열입니다.
- `valueBuf` 인수는 값이 저장되는 버퍼입니다.
- `valueBufLength` 인수는 `GetNoteLength(noteHandle, keyName)`가 반환하는 정수로, 값 버퍼의 최대 길이를 나타냅니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다. `valueBuf` 인수는 키 값을 받습니다.

예제

다음 코드에서는 `welcome.html` 파일과 연관된 디자인 노트 파일에서 `comments` 키 값을 가져옵니다.

```
FileHandle noteHandle = OpenNotesFile("file:///c:/sites/avocado8/iwjs/welcome.html");
if (noteHandle > 0) {
    int valueLength = GetNoteLength(noteHandle, "comments");
    char* valueBuffer = new char[valueLength + 1];
    GetNote(noteHandle, "comments", valueBuffer, valueLength + 1);
    printf("Comments: %s", valueBuffer);
    CloseNotesFile(noteHandle);
}
```

int GetNoteLength()

설명

이 함수는 지정된 키와 연관된 값의 길이를 가져옵니다.

인수

`FileHandle noteHandle`, `const char keyName[64]`

- `noteHandle` 인수는 `OpenNotesFile()` 함수가 반환하는 파일 핸들입니다.
- `keyName[64]` 인수는 키 이름을 포함하는 문자열입니다.

반환값

값의 길이를 나타내는 정수를 반환합니다.

예제

28페이지의 “[BOOL GetNote\(\)](#)”를 참조하십시오.

int GetNotesKeyCount()

설명

이 함수는 지정된 디자인 노트 파일에서 키/값 쌍의 수를 가져옵니다.

인수

FileHandle **noteHandle**

- **noteHandle** 인수는 OpenNotesFile() 함수가 반환하는 파일 핸들입니다.

반환값

디자인 노트 파일에 있는 키/값 쌍의 수를 나타내는 정수를 반환합니다.

BOOL GetNotesKeys()

설명

이 함수는 디자인 노트 파일에 있는 모든 키의 목록을 가져옵니다.

인수

FileHandle **noteHandle**, char* **keyBufArray[64]**, int **keyArrayMaxLen**

- **noteHandle** 인수는 OpenNotesFile()이 반환하는 파일 핸들입니다.
- **keyBufArray[64]** 인수는 키가 저장되는 버퍼 배열입니다.
- **keyArrayMaxLen** 인수는 GetNotesKeyCount(**noteHandle**)가 반환하는 정수로, 키 버퍼 배열의 최대 항목 수를 나타냅니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 true를 반환하고 그렇지 않으면 false를 반환합니다. **keyBufArray** 인수는 키 이름을 받습니다.

예제

다음 코드에서는 welcome.html 파일과 연관된 디자인 노트 파일에 있는 모든 키의 키 이름과 값을 인쇄합니다.

```
typedef char[64] InfoKey;
FileHandle noteHandle = OpenNotesFile("file:///c:/sites/avocado8/iwjs/welcome.html");
if (noteHandle > 0){
    int keyCount = GetNotesKeyCount(noteHandle);
    if (keyCount <= 0)
        return;
    InfoKey* keys = new InfoKey[keyCount];
    BOOL succeeded = GetNotesKeys(noteHandle, keys, keyCount);
    if (succeeded){
        for (int i=0; i < keyCount; i++){
            printf("Key is: %s\n", keys[i]);
            printf("Value is: %s\n\n", GetNote(noteHandle, keys[i]));
        }
    }
    delete []keys;
}
CloseNotesFile(noteHandle);
```

BOOL GetSiteRootForFile()

설명

이 함수는 지정된 디자인 노트 파일의 사이트 루트를 확인합니다.

인수

const char* **filePath**, char* **siteRootBuf**, int **siteRootBufMaxLen**, {InfoPrefs* **infoPrefs**}

- **filePath** 인수는 사이트 루트를 찾을 파일의 file://URL입니다.
- **siteRootBuf** 인수는 사이트 루트가 저장되는 버퍼입니다.
- **siteRootBufMaxLen** 인수는 **siteRootBuf**가 참조하는 버퍼의 최대 크기입니다.
- 선택적인 **infoPrefs** 인수는 사이트의 환경 설정이 저장되는 구조체에 대한 참조입니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 true를 반환하고 그렇지 않으면 false를 반환합니다. **siteRootBuf** 인수는 사이트 루트를 저장하는 버퍼의 주소를 받습니다. **infoPrefs** 인수를 지정하면 이 함수는 사이트의 디자인 노트 환경 설정도 반환합니다. InfoPrefs 구조체에는 bUseDesignNotes 및 bUploadDesignNotes 변수가 있으며 두 변수 모두 BOOL 형식입니다.

BOOL GetVersionName()

설명

이 함수는 MMNotes 공유 라이브러리를 구현한 응용 프로그램을 나타내는 MMNotes 공유 라이브러리 버전 이름을 가져옵니다.

인수

char* **versionNameBuf**, int **versionNameBufMaxLen**

- **versionNameBuf** 인수는 버전 이름이 저장되는 버퍼입니다.
- **versionNameBufMaxLen** 인수는 **versionNameBuf** 인수가 참조하는 버퍼의 최대 크기입니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 true를 반환하고 그렇지 않으면 false를 반환합니다. **versionNameBuf** 인수에 "Dreamweaver"가 저장됩니다.

BOOL GetVersionNum()

설명

이 함수는 특정 함수가 사용 가능한지 여부를 확인하는 데 사용할 수 있는 MMNotes 공유 라이브러리의 버전 번호를 가져옵니다.

인수

char* **versionNumBuf**, int **versionNumBufMaxLen**

- **versionNumBuf** 인수는 버전 번호가 저장되는 버퍼입니다.
- **versionNumBufMaxLen** 인수는 **versionNumBuf**가 참조하는 버퍼의 최대 크기입니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다. **versionNumBuf** 인수는 버전 번호를 저장합니다.

BOOL LocalURLToFilePath()

설명

이 함수는 지정된 **file://** URL을 로컬 드라이브 경로로 변환합니다.

인수

const char* localURL, char* drivePathBuf, int drivePathMaxLen

- **localURL** 인수는 로컬 파일의 경로이며, **file://** URL로 표시됩니다.
- **drivePathBuf** 인수는 로컬 드라이브 경로가 저장되는 버퍼입니다.
- **drivePathMaxLen** 인수는 **drivePathBuf** 인수가 참조하는 버퍼의 최대 크기입니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다. **drivePathBuf** 인수는 로컬 드라이브 경로를 받습니다.

FileHandle OpenNotesFile()

설명

이 함수는 지정된 파일과 연관된 디자인 노트 파일을 열거나, 해당 디자인 노트 파일이 없는 경우 새로 만듭니다.

인수

const char* localFileURL, {BOOL bForceCreate}

- **localFileURL** 인수는 디자인 노트 파일이 연관된 주 파일의 경로를 포함하는 문자열이며, **file://** URL로 표시됩니다.
- **bForceCreate** 인수는 디자인 노트가 사이트에 대해 해제되어 있거나 **localFileURL** 인수로 지정한 경로가 사이트와 연관되어 있지 않은 경우에도 디자인 노트 파일을 만들지 여부를 나타내는 부울 값입니다.

FileHandle OpenNotesFilewithOpenFlags()

설명

이 함수는 지정된 파일과 연관된 디자인 노트 파일을 열거나, 해당 디자인 노트 파일이 없는 경우 새로 만듭니다. 파일을 읽기 전용 모드로 열 수 있습니다.

인수

const char* localFileURL, {BOOL bForceCreate}, {BOOL bReadOnly}

- **localFileURL** 인수는 디자인 노트 파일이 연관된 주 파일의 경로를 포함하는 문자열이며, **file://** URL로 표시됩니다.
- **bForceCreate** 인수는 디자인 노트가 사이트에 대해 해제되어 있거나 경로가 사이트와 연관되어 있지 않은 경우에도 디자인 노트 파일을 만들지 여부를 나타내는 부울 값입니다. 기본값은 **false**입니다. 이 인수는 선택 사항이지만 세 번째 인수를 지정한 경우에는 반드시 지정해야 합니다.

- 선택적인 **bReadOnly** 인수는 읽기 전용 모드로 파일을 열지 여부를 나타내는 부울 값입니다. 기본값은 **false**입니다. MMNotes.dll 파일 버전 2부터 **bReadOnly** 인수를 지정할 수 있습니다.

BOOL RemoveNote()

설명

이 함수는 지정된 디자인 노트 파일에서 지정된 키와 그 값을 제거합니다.

인수

FileHandle noteHandle, const char keyName[64]

- **noteHandle** 인수는 OpenNotesFile() 함수가 반환하는 파일 핸들입니다.
- **keyName[64]** 인수는 제거할 키 이름을 포함하는 문자열입니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

BOOL SetNote()

설명

이 함수는 디자인 노트 파일에서 키/값 쌍 하나를 만들거나 업데이트합니다.

인수

FileHandle noteHandle, const char keyName[64], const char* value

- **noteHandle** 인수는 OpenNotesFile() 함수가 반환하는 파일 핸들입니다.
- **keyName[64]** 인수는 키 이름을 포함하는 문자열입니다.
- **value** 인수는 값을 포함하는 문자열입니다.

반환값

부울 값을 반환합니다. 작업이 성공하면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

5장: Fireworks 통합

FWLaunch는 객체, 명령, 비헤이비어 및 속성 관리자의 제작자가 Adobe® Fireworks®와 통신할 수 있도록 하는 C 공유 라이브러리입니다. FWLaunch를 사용하면 JavaScript를 작성하여 Fireworks 사용자 인터페이스를 열고 **Fireworks** 확장에 설명되어 있는 고유의 JavaScript API를 통해 Fireworks에 명령을 제공할 수 있습니다. Adobe® Dreamweaver® CS4에서 C 라이브러리가 JavaScript 인터프리터와 상호 작용하는 방법에 대한 일반적인 내용은 **Dreamweaver** 확장에서 C 레벨 확장성에 대한 정보를 참조하십시오.

FWLaunch API

FWLaunch 객체는 Extension에서 Fireworks를 열고 Fireworks JavaScript API를 사용하여 Fireworks 작업을 수행한 다음 값을 Dreamweaver에 반환할 수 있도록 합니다. 이 장에서는 FWLaunch 통신 API와 그 사용 방법에 대해 설명합니다.

FWLaunch.bringDWToFront()

지원 버전

Dreamweaver 3, Fireworks 3

설명

이 함수는 Dreamweaver를 맨 앞으로 가져옵니다.

인수

없음

반환값

없음

FWLaunch.bringFWToFront()

지원 버전

Dreamweaver 3, Fireworks 3

설명

이 함수는 Fireworks가 실행 중일 경우 맨 앞으로 가져옵니다.

인수

없음

반환값

없음

FWLaunch.execJsInFireworks()

지원 버전

Dreamweaver 3, Fireworks 3

설명

이 함수는 실행할 Fireworks로 지정된 JavaScript 또는 JavaScript 파일에 대한 참조를 전달합니다.

인수

javascriptOrFileURL

- javascriptOrFileURL 인수는 리터럴 JavaScript 문자열 또는 JavaScript 파일의 경로이며, file:// URL로 표시됩니다.

반환값

JavaScript가 성공적으로 전달되면 쿠키 객체를 반환하고 다음 오류 중 하나가 발생하면 이를 나타내는 0이 아닌 오류 코드를 반환합니다.

- 잘못된 사용 - javascriptOrFileURL 인수가 null 값 또는 빈 문자열로 지정되었거나 JS 또는 JSF 파일의 경로가 잘못된 경우
- 파일 I/O 오류 - 디스크가 꽉 차서 Fireworks에서 응답 파일을 만들 수 없는 경우
- Dreamweaver 오류 - 사용자가 유효한 Dreamweaver 버전(버전 3 이상)을 실행하고 있지 않은 경우
- Fireworks 프로세스 시작 오류 - 함수가 유효한 Fireworks 버전(버전 3 이상)을 열지 않는 경우
- 사용자가 작업을 취소한 경우

FWLaunch.getJsResponse()

지원 버전

Dreamweaver 3, Fireworks 3

설명

이 함수는 FWLaunch.execJsInFireworks() 함수가 전달한 JavaScript를 Fireworks가 아직 실행하고 있는지, 아니면 스크립트가 성공적으로 완료되었는지, 아니면 오류가 발생했는지를 확인합니다.

인수

progressTrackerCookie

- progressTrackerCookie 인수는 FWLaunch.execJsInFireworks() 함수가 반환하는 쿠키 객체입니다.

반환값

작업이 성공적으로 완료되면 FWLaunch.execJsInFireworks() 함수에 전달된 스크립트의 결과가 포함된 문자열을 반환하고, Fireworks가 아직 JavaScript를 실행 중이면 null 값을 반환하며, 다음 오류 중 하나가 발생했으면 이를 나타내는 0이 아닌 오류 코드를 반환합니다.

- 잘못된 사용 - Fireworks에서 스크립트를 실행할 때 JavaScript 오류가 발생한 경우
- 파일 I/O 오류 - 디스크가 꽉 차서 Fireworks에서 응답 파일을 만들 수 없는 경우
- Dreamweaver 오류 - 사용자가 유효한 Dreamweaver 버전(버전 3 이상)을 실행하고 있지 않은 경우
- Fireworks 프로세스 시작 오류 - 함수가 유효한 Fireworks 버전(버전 3 이상)을 열지 않는 경우
- 사용자가 작업을 취소한 경우

예제

다음 코드에서는 "prompt('Please enter your name:')" 문자열을 FWLaunch.execJsInFireworks()에 전달하고 결과를 확인합니다.

```
var progressCookie = FWLaunch.execJsInFireworks("prompt('Please enter your name:')");
var doneFlag = false;
while (!doneFlag){
    // check for completion every 1/2 second
    setTimeout('checkForCompletion()',500);
}
function checkForCompletion(){
    if (progressCookie != null) {
        var response = FWLaunch.getJsResponse(progressCookie);
        if (response != null) {
            if (typeof(response) == "number") {
                // error or user-cancel, time to close the window
                // and let the user know we got an error
                window.close();
                alert("An error occurred.");
            }else{
                // got a valid response!
                alert("Nice to meet you, " + response);
                window.close();
            }
        }
        doneFlag = true;
    }
}
```

FWLaunch.mayLaunchFireworks()

지원 버전

Dreamweaver 2, Fireworks 2

설명

이 함수는 Fireworks 최적화 세션을 열 수 있는지 여부를 확인합니다.

인수

없음

반환값

플랫폼이 Windows인지, 아니면 Macintosh인지를 나타내는 부울 값을 반환합니다. Macintosh인 경우 다른 Fireworks 최적화 세션이 이미 실행 중인지 여부를 나타냅니다.

FWLaunch.optimizeInFireworks()

지원 버전

Dreamweaver 2, Fireworks 2

설명

이 함수는 지정된 이미지에 대해 Fireworks 최적화 세션을 엽니다.

인수**docURL, imageURL, {targetWidth}, {targetHeight}**

- **docURL** 인수는 활성 문서의 경로이며, `file://` URL로 표시됩니다.
- **imageURL** 인수는 선택된 이미지의 경로입니다. 상대 경로인 경우 **docURL** 인수에 지정하는 경로에 대해 상대적입니다.
- 선택적인 **targetWidth** 인수는 이미지의 크기를 조절할 폭을 정의합니다.
- 선택적인 **targetHeight** 인수는 이미지의 크기를 조절할 높이를 정의합니다.

반환값

지정된 이미지에 대해 Fireworks 최적화 세션이 성공적으로 열리면 0을 반환하고 다음 오류 중 하나가 발생했으면 이를 나타내는 0이 아닌 오류 코드를 반환합니다.

- 잘못된 사용 - **docURL** 인수, **imageURL** 인수 또는 두 인수 모두가 null 값 또는 빈 문자열로 지정된 경우
- 파일 I/O 오류 - 디스크가 꽉 차서 Fireworks에서 응답 파일을 만들 수 없는 경우
- Dreamweaver 오류 - 사용자가 유효한 Dreamweaver 버전(버전 2 이상)을 실행하고 있지 않은 경우
- Fireworks 프로세스 시작 오류 - 함수가 유효한 Fireworks 버전(버전 2 이상)을 열지 않는 경우
- 사용자가 작업을 취소한 경우

FWLaunch.validateFireworks()**지원 버전**

Dreamweaver 2, Fireworks 2

설명

이 함수는 사용자의 하드 디스크에서 지정된 Fireworks 버전을 찾습니다.

인수**{versionNumber}**

- **versionNumber** 인수는 2보다 크거나 같은 선택적인 부동 소수점 수로, 필요한 Fireworks의 버전을 나타냅니다. 이 인수를 생략하면 기본값은 2입니다.

반환값

지정된 버전의 Fireworks를 발견했는지 여부를 나타내는 부울 값을 반환합니다.

예제

다음 코드에서는 Fireworks가 설치되어 있는지 여부를 확인합니다.

```
if (FWLaunch.validateFireworks(6.0)){
    alert("Fireworks 6.0 or later is installed.");
}else{
    alert("Fireworks 6.0 is not installed.");
}
```

간단한 FWLaunch 통신 예제

다음 명령은 Fireworks가 사용자에게 이름을 물어본 후 해당 이름을 Dreamweaver에 반환합니다.

```
<html>
<head>
<title>Prompt in Fireworks</title>
<meta http-equiv="Content-Type" content="text/html; ~
charset=iso-8859-1">
<script>

function commandButtons(){
    return new Array("Prompt", "promptInFireworks()", "Cancel", ~
    "readyToCancel()", "Close","window.close()");
}

var gCancelClicked = false;
var gProgressTrackerCookie = null;

function readyToCancel() {
    gCancelClicked = true;
}

function promptInFireworks() {
    var isFireworks3 = FWLaunch.validateFireworks(3.0);
    if (!isFireworks3) {
        alert("You must have Fireworks 3.0 or later to use this ~
        command");
    }
    return;

    // Tell Fireworks to execute the prompt() method.
    gProgressTrackerCookie = FWLaunch.execJsInFireworks~
    ("prompt('Please enter your name:')");

    // null means it wasn't launched, a number means an error code
    if (gProgressTrackerCookie == null || ~
    typeof(gProgressTrackerCookie) == "number") {
        window.close();
        alert("an error occurred");
        gProgressTrackerCookie = null;
    } else {
        // bring Fireworks to the front
        FWLaunch.bringFWToFront();
        // start the checking to see if Fireworks is done yet
        checkOneMoreTime();
    }
}

function checkOneMoreTime() {
    // Call checkJsResponse() every 1/2 second to see if Fireworks
    // is done yet
    window.setTimeout("checkJsResponse();", 500);
}

function checkJsResponse() {
    var response = null;

    // The user clicked the cancel button, close the window
    if (gCancelClicked) {
        window.close();
        alert("cancel clicked");
    } else {
        // We're still going, ask Fireworks how it's doing
        if (gProgressTrackerCookie != null)
            response = FWLaunch.getJsResponse(gProgressTrackerCookie);
    }
}
```

```
if (response == null) {
    // still waiting for a response, call us again in 1/2 a
    // second
    checkOneMoreTime();
} else if (typeof(response) == "number") {
    // if the response was a number, it means an error occurred
    // the user cancelled in Fireworks
    window.close();
    alert("an error occurred.");
} else {
    // got a valid response! This return value might not
    // always be a useful one, since not all functions in
    // Fireworks return a string, but we know this one does,
    // so we can show the user what we got.
    window.close();
    FWLaunch.bringDWToFront();// bring Dreamweaver to the front
    alert("Nice to meet you, " + response + "!");
}
}
</script>
</head>
<body>
<form>
<table width="313" nowrap>
<tr>
<td>This command asks Fireworks to execute the prompt() ~
function. When you click Prompt, Fireworks comes forward and ~
asks you to enter a value into a dialog box. That value is then ~
returned to Dreamweaver and displayed in an alert.</td>
</tr>
</table>
</form>
</body>
</html>
```

6장: Flash 통합

Adobe® Dreamweaver®는 Flash Generator 템플릿 파일을 사용하여 새 Flash 객체를 만드는 Flash 객체 API에 대한 지원을 제공합니다. Flash 객체 API 항목에서는 Flash Generator 템플릿(SWT 파일)에서 Flash 객체(SWF 파일)를 만드는 방법에 대해 설명합니다.

Flash 내용을 Dreamweaver 객체 또는 명령에 추가하는 방법에 대한 자세한 내용은 **Dreamweaver** 확장을 참조하십시오.

Flash 객체 API

Extension 개발자는 Flash 객체 API를 사용하여 Flash Generator를 통해 간단한 SWF 파일을 만드는 객체를 구성할 수 있습니다. 이 API를 사용하여 Flash Generator 템플릿에서 매개 변수를 설정하고 SWF 파일이나 이미지 파일을 출력할 수 있습니다. 또한 기존 Flash 객체를 읽고 조작할 수 있을 뿐 아니라 새 Flash 객체를 만들 수도 있습니다.

SWT 파일은 Flash Generator 템플릿 파일로서, Flash 객체 파일을 구성하는 데 필요한 모든 정보를 포함합니다. 이러한 API 함수를 사용하여 SWT 파일에서 SWF 파일이나 이미지 파일을 만들 수 있습니다. SWF 파일은 SWT 파일의 매개 변수를 실제 값으로 바꾸어 만들어집니다. Flash에 대한 자세한 내용은 Flash 설명서를 참조하십시오. 다음 함수는 SWFFile 객체의 메서드입니다.

SWFFile.createFile()

설명

이 함수는 지정된 템플릿과 매개 변수 배열을 사용하여 새 Flash 객체 파일을 생성합니다. 또한 GIF, PNG, JPEG 및 MOV 형식의 파일 이름이 지정되면 해당 제목의 GIF, PNG, JPEG 및 MOV 버전도 만듭니다.

지정하려는 선택적 매개 변수 앞에 포함하지 않을 선택적 매개 변수가 오는 경우 사용하지 않을 매개 변수는 빈 문자열로 지정해야 합니다. 예를 들어, GIF 파일이 아닌 PNG 파일을 지정하려면 PNG 파일 이름을 지정하기 전에 빈 문자열을 지정해야 합니다.

인수

templateFile, templateParams, swfFileName, {gifFileName}, {pngFileName}, {jpgFileName}, {movFileName}, {generatorParams}

- **templateFile** 인수는 템플릿 파일의 경로이며, file:// URL로 표시됩니다. 이 파일은 SWT 파일일 수 있습니다.
- **templateParams** 인수는 이름/값 쌍의 배열이며, 여기서 이름은 SWT 파일의 매개 변수를 나타내고 값은 이러한 매개 변수에 지정할 값입니다. Dreamweaver에서 SWF 파일을 Flash 객체로 인식하도록 하려면 첫 번째 매개 변수가 "dwType"이어야 합니다. 그리고 그 값은 "Flash Text"와 같이 객체 유형의 이름을 나타내는 문자열이어야 합니다.
- **swfFileName** 인수는 SWF 파일의 출력 파일 이름이며, file:// URL로 표시됩니다. 빈 문자열인 경우에는 무시됩니다.
- **gifFileName** 인수는 GIF 파일의 출력 파일 이름이며, file:// URL로 표시됩니다. 이 인수는 선택 사항입니다.
- **pngFileName** 인수는 PNG 파일의 출력 파일 이름이며, file:// URL로 표시됩니다. 이 인수는 선택 사항입니다.
- **jpgFileName** 인수는 JPEG 파일의 출력 파일 이름이며, file:// URL로 표시됩니다. 이 인수는 선택 사항입니다.
- **movFileName** 인수는 QuickTime 파일의 출력 파일 이름이며, file:// URL로 표시됩니다. 이 인수는 선택 사항입니다.
- **generatorParams** 인수는 선택적 Generator 명령줄 플래그를 나타내는 문자열 배열입니다. 이 인수는 선택 사항입니다. 각 플래그의 데이터 항목은 배열에서 플래그 뒤에 나열되어야 합니다. 다음 표에서는 일반적으로 사용되는 몇 가지 플래그를 보여 줍니다.

옵션 플래그	데이터	설명	예제
-defaultsize	폭, 높이	출력 이미지 크기를 지정된 폭과 높이로 설정합니다.	"-defaultsize", "640", "480"
-exactFit	없음	출력 이미지의 내용을 확장하여 지정된 출력 크기에 정확히 맞춥니다.	"-exactFit"

반환값

다음 값 중 하나가 들어 있는 문자열을 반환합니다.

- "noError"는 호출이 성공적으로 끝났음을 나타냅니다.
- "invalidTemplateFile"은 지정된 템플릿 파일이 잘못되었거나 없음을 나타냅니다.
- "invalidOutputFile"은 지정된 출력 파일 이름 중 하나 이상이 잘못되었음을 나타냅니다.
- "invalidData"는 **templateParams** 인수의 이름/값 쌍 중 하나 이상이 잘못되었음을 나타냅니다.
- "initGeneratorFailed"는 **Generator**를 초기화할 수 없음을 나타냅니다.
- "outOfMemory"는 메모리가 부족하여 작업을 완료할 수 없음을 나타냅니다.
- "unknownError"는 알 수 없는 오류가 발생했음을 나타냅니다.

예제

다음 JavaScript에서는 템플릿 파일 내의 모든 "text" 문자열을 "Hello World" 문자열로 바꾸는 "myType" 유형의 Flash 객체 파일을 만듭니다. 여기서는 SWF 파일 뿐만 아니라 GIF 파일도 만듭니다.

```
var params = new Array;
params[0] = "dwType";
params[1] = "myType";
params[2] = "text";
params[3] = "Hello World";
errorString = SWFFile.createFile( "file:///MyMac/test.swt", ~
params, "file:///MyMac/test.swf", "file:///MyMac/test.gif");
```

SWFFile.getNaturalSize()

설명

이 함수는 압축되지 않은 Flash 내용의 원래 크기를 반환합니다.

인수

fileName

- **fileName** 인수는 Flash 내용의 경로이며, file:// URL로 표시됩니다.

반환값

압축되지 않은 SWF 파일의 폭과 높이를 나타내는 두 요소가 들어 있는 배열을 반환하거나, 압축되지 않은 SWF 파일이 아닌 경우에는 null 값을 반환합니다.

SWFFile.getObjectType()

설명

이 함수는 Flash 객체 유형, 즉 SWFFile.createFile() 함수에서 파일을 만들었을 때 dwType 매개 변수로 전달된 값을 반환합니다.

인수**fileName**

- **fileName** 인수는 Flash 객체 파일의 경로이며, file:// URL로 표시됩니다. 이 파일은 대개 SWF 파일입니다.

반환값

객체 유형을 나타내는 문자열을 반환하거나, Flash 객체 파일이 아니거나 파일을 찾을 수 없을 경우에는 null을 반환합니다.

예제

다음 코드에서는 test.swf 파일이 myType 유형의 Flash 객체인지 여부를 확인합니다.

```
if ( SWFFile.getObjectType("file:///MyMac/test.swf") == "myType" ){
    alert ("This is a myType object.");
}else{
    alert ("This is not a myType object.");
}
```

SWFFile.readFile()

설명

이 함수는 Flash 객체 파일을 읽습니다.

인수**fileName**

- **fileName** 인수는 Flash 객체 파일의 경로이며, file:// URL로 표시됩니다.

반환값

첫 번째 배열 요소가 템플릿 SWT 파일의 전체 경로인 문자열 배열을 반환합니다. 다음 문자열은 객체의 매개 변수(이름/값 쌍)를 나타냅니다. 배열에서 각 이름 뒤에는 값이 옵니다. 첫 번째 이름/값 쌍은 "dwType"과 그 값입니다. 이 함수는 파일을 찾을 수 없거나 Flash 객체 파일이 아닌 경우 null 값을 반환합니다.

예제

var params = SWFFile.readFile("file:///MyMac/test.swf")을 호출하면 매개 변수 배열에 다음 값이 반환됩니다.

```
"file:///MyMac/test.swt"    // template file used to create this .swf file
"dwType"                   // first parameter
"myType"                   // first parameter value
"text"                     // second parameter
"Hello World"              // second parameter value
```

Flash 패널 및 대화 상자 함수

다음 API를 사용하여 패널과 대화 상자에서 SWF 파일을 추가할 수 있습니다.

dreamweaver.flash.newControl()

지원 버전

Dreamweaver CS4

설명

이 함수를 사용하면 Flash 컨트롤을 만들 수 있습니다. 이 컨트롤은 나중에 controlID 매개 변수를 통해 참조됩니다. 이 컨트롤은 SWF 경로로 지정된 Flash 파일(.swf)을 표시합니다. 이 컨트롤은 배치되고 defaultGeometry 매개 변수에 지정된 크기를 갖습니다.

참고: Dreamweaver에서는 flash.requestStateChange를 호출하는 경우 Flash 컨트롤을 표시하고, newControl을 호출하는 경우 Dialog 컨트롤을 표시합니다. flash.requestStateChange를 호출할 필요는 없습니다.

인수

controlID, controlType, controlData

- controlID 인수는 문자열 값입니다.
- controlType 인수는 패널이 표준 Extension("standard"), 신뢰할 수 있는 표준 Extension("trusted") 또는 추가 Extension(다른 임의의 값) 중에서 어느 것인지 지정합니다. 추가 Extension인 경우 해당 값은 필요한 사용자 정의 통합의 유형을 나타내는 호스트 응용 프로그램에 특별히 알려진 식별자입니다. 응용 프로그램에서 사용자 정의 통합 유형을 인식하지 못하면 오류가 반환됩니다.
- controlData는 객체입니다. 이 인수의 주요 속성은 다음과 같습니다.

속성	설명	값
controlData.swfUTF8Path	SWF의 위치. 이 속성은 필수이며 JavaScript의 모든 문자가 유니코드로 되어 있으므로 유니코드 문자열로 전달됩니다.	controlData.windowType에 사용 가능한 값 - PanelWindow.이 표 다음에 나오는 표에 이 값에 대한 사양이 나와 있습니다. - ModalDialogWindow
{controlData.scriptPath}	외부 인터페이스 호출을 사용하여 .swf에서 실행할 함수가 포함된 .js 파일의 경로. 이 속성은 선택 사항입니다. 외부 인터페이스를 사용하여 .swf 파일에서 Dreamweaver의 JavaScript 코드로 콜백하려면 .swf 파일에서 호출할 수 있는 함수가 포함된 .js 파일을 제공할 수 있습니다. 자세한 내용은 dw.flash.executeScript 호출을 참조하십시오.	
controlData.defaultGeometry	defaultGeometry 값은 화면 왼쪽 상단에서의 화면 좌표로 표현됩니다. 이 속성은 필수입니다. <pre>Object /*!< default creation geometry, including positioning */ { topleftx: Number, toplefty: Number, width: Number, height: Number }</pre>	

다음 표에서는 PanelWindow 사양을 보여 줍니다.

옵션	유형	설명
name	문자열	템에 나타나는 패널의 이름. 이 옵션을 지정하지 않으면 "UNDEFINED"로 명명됩니다. 모든 패널 이름은 대문자로 나타납니다. 이 이름을 소문자로 변경할 수 없습니다.
{controlData.minSize}	객체	<i>minSize</i> 는 PanelWindow 유형의 컨트롤에만 적용됩니다. 이 옵션은 패널의 크기를 조절할 수 있는 최소 크기를 제어합니다. 이 옵션은 선택 사항입니다. <i>minSize</i> 가 지정되지 않은 경우 기본값은 <i>defaultGeometry</i> 에 지정된 폭과 높이이고 패널의 크기를 조절할 수 없습니다. { width: Number, height: Number }
{controlData.maxSize}	객체	<i>maxSize</i> 는 PanelWindow 유형의 컨트롤에만 적용됩니다. 이 옵션은 선택 사항입니다. 이 옵션은 패널의 크기를 조절할 수 있는 최대 크기를 제어합니다. <i>maxSize</i> 가 지정되지 않은 경우 기본값은 <i>defaultGeometry</i> 에 지정된 폭과 높이이고 패널의 크기를 조절할 수 없습니다. { width: Number, height: Number }
{iconPathNormal}	문자열	패널이 아이콘 모드로 축소될 때 부동 패널에서 사용되어야 하는 아이콘의 경로. 이 옵션은 선택 사항입니다.
{iconPathRollOver}	문자열	패널이 아이콘 모드로 축소되고 사용자가 그 위에 포인터를 놓을 때 부동 패널에서 사용되어야 하는 아이콘의 경로. 이 옵션은 선택 사항입니다.
{iconPathDisable}	문자열	패널이 아이콘 모드로 축소되고 비활성화될 때 부동 패널에서 사용되어야 하는 아이콘의 경로. 이 옵션은 선택 사항입니다.

반환값

다음 성공 또는 오류 코드 중 하나를 반환합니다.

- PlugPlugErrorCode_success 코드는 컨트롤 생성에 성공했는지 여부를 알려줍니다.
- PlugPlugErrorCode_extensionRegistrationFailed 코드는 컨트롤을 등록할 수 없다는 것을 알려줍니다.

dreamweaver.flash.requestStateChange()

지원 버전

Dreamweaver CS4

설명

이 함수는 extensionID를 가진 Extension에 대해 *uniqueID*로 식별된 부동 패널의 상태를 변경합니다.

인수

controlID, *stateChange*, *stateData*

- controlID 인수는 문자열 값입니다.
- stateChange 인수는 사용 가능한 다음 값이 포함된 문자열입니다.

값	설명
Move	크기를 제외하고 원점만 변경됨
Resize	크기가 변경되고 원점도 변경될 수 있음
Show	표시되지만 하고 크기와 위치는 변경되지 않음
Hide	숨기기만 하고 크기와 위치는 변경되지 않음
Minimize	Hide와 유사하지만 숨겨지는 이유가 설명됨
Restore	Show와 유사하지만 표시되는 이유가 설명됨
Open	윈도우가 만들어지고 Extension이 로드됨
Close	포함된 Extension이 로드되지 않음

- stateData 인수의 값은 다음 표에 표시된 문자열입니다.

stateChange의 값	stateData의 값
Move	eventData = { topleftx: Number, toplefty: Number }
Resize	eventData = { width: Number, height: Number }

반환값

다음 표에는 문자열인 반환값이 포함되어 있습니다.

값	설명
RequestPosted	요청을 실행할 이벤트나 명령이 호스트 응용 프로그램에서 대기되었습니다.
RequestComplete	호스트 응용 프로그램에서 요청을 성공적으로 완료했습니다.
RequestFailed	호스트 응용 프로그램에서 요청을 완료하려고 했지만 실패했습니다.
RequestDenied	호스트 응용 프로그램에서 요청된 작업을 지원하지 않기 때문에 요청을 거부했습니다.

예제

```
controlData = {};
controlData.defaultGeometry = {topleftx : 100, toplefty : 100, width : 200, height : 200 };
controlData.minSize = {width : 100; height : 100 };
controlData.maxSize = {width : 300; height : 300 };
var swfPath = dw.getConfigurationPath();
swfPath += '/flash/PhotoAlbum.swf';
controlData.swfUTF8Path = swfPath;
// open the window
flash.requestStateChange("com.adobe.extension.foo","Open",controlData.defaultGeometry);
```

dreamweaver.flash.controlEvent()

지원 버전

Dreamweaver CS4

설명

이 함수는 Flash 컨트롤에 이벤트를 전달하는 데 사용됩니다. 이벤트 호출은 함수와 관련 매개 변수를 캡처하는 XML 문자열로 전달됩니다. XML 문자열은 SWF 파일에서 시작되어야 하는 함수를 캡처합니다.

인수

inControlID, inXMLString

- inControlID 인수는 문자열입니다.
- inXMLString 인수는 문자열입니다. 다음 inXMLString을 전달하여 flashCallback Flash 파일에서 함수를 호출하고 단일 문자열 'Hello'를 인수로 전달합니다.

```
<invoke name="flashCallback" returntype="xml">
  <arguments>
    <string>Hello</string>
  </arguments>
</invoke>
```

반환값

XML 문자열을 반환합니다.

예제

다음 예제에서는 JavaScript에서 flashCallback 함수를 호출합니다. 이 예제에서는 콜백 함수 이름 및 인수를 XML 문자열로 전달합니다.

```
var xmlString = '<invoke name="flashCallback" returntype="xml">
<arguments>
<string>Hello</string>
</arguments>
</invoke>';
```

또한 dw.flash.controlEvent를 사용하여 Flash 파일(.swf)로 콜백합니다.

```
dw.flash.controlEvent('Flickr', xmlString);
```

이 함수에서 사용되는 인수는 다음과 같습니다.

- .swf 컨트롤이 dw.flash.newControl을 사용하여 만들어질 때 전달된 Extension의 ID인 Flickr
- 콜백 함수와 인수가 포함된 XML 문자열

다음 예제는 flashcallback.mxml에서 구현된 flashcallback 함수의 구현입니다. 다음 예제에서는 flashcallback 함수를 추가합니다. 이 함수는 외부 응용 프로그램에서 호출되어야 합니다.

```
public function initApp():void {
    ExternalInterface.addCallback("flashCallback",flashCallback);
}
```

이 함수는 Flash 파일(.swf) 외부에서 콜백됩니다.

참고: 이 함수를 호출하기 전에 ExternalInterface.addCallback("flashCallback",flashCallback)을 호출해야 합니다.

```
public function flashCallback(inputStr:String):String
{
    out.text += inputStr + " got flashCallback!\n";
    return "it worked!";
}
```

dreamweaver.flash.setMenu()

지원 버전

Dreamweaver CS4

설명

이 함수를 사용하면 "PanelWindow" 유형의 Extension에 대한 플라이아웃 명령을 제공할 수 있습니다.

인수

inControlID, inMenuPosition, inMenu

- *inControlID*는 **extension ID**입니다. 이 함수를 호출하면 **Extension**이 포함된 열린 패널의 [플라이아웃] 메뉴가 영향을 받습니다. 이 인수가 정의되어 있지 않은 경우 함수를 호출하면 응용 프로그램의 기본 메뉴가 영향을 받습니다.
- *inMenuPosition*은 지정된 명령을 배치해야 하는 위치를 설명하는 문자열입니다.
 - 이 문자열이 정의되어 있지 않으면 전체 메뉴가 대체됩니다.
 - 이 문자열이 패널용이면 사용자가 설정할 수 있는 [플라이아웃] 메뉴의 전체 영역이 대체됩니다. 응용 프로그램에서는 고정된 일부 플라이아웃 항목을 예약합니다.
 - 이 문자열이 응용 프로그램용이면 [윈도우] 메뉴의 전체 기본 [컨트롤] 하위 메뉴가 대체됩니다.
 - 이 문자열이 메뉴의 섹션을 설정하기 위해 결정될 스키마에 있는 **XML** 문자열이면 이 양식이 이후 호환성을 위해 제공됩니다.
- *inMenu*는 **MenuItem**과 동일한 기능을 수행합니다. 이 인수는 지정된 메뉴 위치에 추가되는 명령의 목록을 나타내며 이전 호출을 통해 해당 위치에 추가된 이전 항목을 대체합니다.

반환값

다음 성공 또는 오류 코드 중 하나를 반환합니다.

- **PlugPlugErrorCode_success** 코드는 성공을 나타냅니다.
- **PlugPlugErrorCode_extensionMenuCreationFailed** 코드는 **Extension** 메뉴를 만들지 못했음을 나타냅니다.
- **PlugPlugErrorCode_unknown** 코드는 함수가 알 수 없는 이유로 실패했음을 나타냅니다.

예제

다음 예제는 메뉴를 설정하는 데 사용됩니다.

```
function initializeMenuItem(menuID, menuName,extensionID, submenu)
{
    var menuItem = {};
    menuItem.menuID = menuID; //!< unique menu ID, if NULL menu is disabled
    menuItem.nameUtf8 = menuName; //!< Item title, if "---" item is a separator
    menuItem.extensionID = extensionID; //!< optional extension ID, used for panels only
    menuItem.submenu = submenu; //!< if non-NULL, this is a submenu
    return menuItem;
}

function setupMenu()
{
    var menuItems = new Array();
    menuItems.push(initializeMenuItem('id1','Call .swf
        ActionScript',undefined,undefined));
    menuItems.push(initializeMenuItem('id0','---',undefined,undefined));
    menuItems.push(initializeMenuItem('id2','Call Dw JavaScript',undefined,undefined));
    dw.flash.setMenu('Flickr',controlID,menuItems);
}
```

참고: **newControl**에 전달된 객체의 **scriptPath**에 지정된 **JavaScript** 파일에서 **"onSelectMenuItem"**이라는 함수를 지정하십시오.

onSelectMenuItem은 메뉴 항목 핸들러이며 명령이 플로터의 [플라이아웃] 메뉴에서 선택되면 해당 메뉴 ID를 사용하여 호출됩니다.

다음 예제에서는 'Configuration/flash/Flickr.js'에서 콜백 핸들러 정의를 지정합니다.

```

        function onSelectMenuItem(menuID)
        {
            if (menuID == 'idl1') {
                var flashCallbackString = '<invoke name= " flash Callback"
                returntype="xml">
                <arguments><string>Hello</string></arguments></invoke>';
                dw.flash.control Event('Flickr', flashCallbackString);
                return("PlugPlugRequestCompleted");
            } else {
                alert ( ' You selected: menuID = ' + menuID);
                return ( " PlugPlugRequestCompleted");
            }
        }
    }

```

dreamweaver.flash.evalScript()

지원 버전

Dreamweaver CS4

설명

이 함수는 다음과 같은 목적으로 JavaScript 함수를 호출하는 데 사용됩니다.

- CSXS Extension의 Extension과 연결된 스크립트 파일에 정의된 JavaScript 함수 실행
- CSXS 기반이 아닌 Extension에 대한 scriptPath 매개 변수에 정의된 .js 파일

인수

controlID, javascript function call

- controlID 인수는 스크립트를 실행할 Extension의 ID입니다. 이 ID는 dw.flash.newControl()의 첫 번째 매개 변수로 지정된 ID와 일치해야 합니다.
- JavaScript 함수 호출 인수는 사용자가 원하는 수의 매개 변수를 사용하여 함수를 호출할 수 있도록 합니다.

반환값

부울 값을 반환합니다. 함수가 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.flash.executeScript()

지원 버전

Dreamweaver CS4

설명

이 함수는 .js 파일의 함수를 실행하는 데 사용됩니다. .swf 파일의 ActionScript는 dreamweaver.flash.executeScript() 함수를 시작합니다.

인수

javascript function call

참고: 호출할 함수가 포함된 .js 파일의 경로를 지정하십시오.

반환값

ActionScript 객체로 직렬화되는 XML 문자열을 반환합니다.

예제

다음 예제에는 샘플 파일 **Sample.mxml**과 JavaScript 파일 **Sample.js**의 JavaScript 함수가 포함되어 있습니다.

```
private function executeScript():void
{
    if(ExternalInterface.available)
    {
        out.text += "SwfCalledHost\n";
        var scriptText:String = "helloWorld('scott');\n";
        var resultStr:Object =
            ExternalInterface.call("dw.flash.executeScript",scriptText);
        out.text += "Result: " + resultStr.strResult + '\n';
    }
}
```

다음 JavaScript 파일에는 .swf에서 호출되는 JavaScript 함수 **helloWorld()**가 포함되어 있습니다. 이 함수에서는 **dw.getAppLanguage()** 호출을 사용하여 Dreamweaver가 **Sample.js**에서 실행 중인 5자로 된 언어 코드를 반환합니다.

```
function helloWorld(nameStr)
{
    alert('hello ' + nameStr);
    var appLanguage = dw.getAppLanguage();
    var returnStr = '<object><property id="strResult"><string>Language: ' + appLanguage
        + '</string></property></object>';
    alert(returnStr);
    return (returnStr);
}
```

참조

41페이지의 “[dreamweaver.flash.newControl\(\)](#)”

dreamweaver.flash.controlExists

지원 버전

Dreamweaver CS4

설명

이 함수는 컨트롤이 있는지 확인하는 데 사용됩니다. **PanelWindow** 컨트롤은 **Dreamweaver** 시작 사이에 저장됩니다.

인수

controlID

반환값

부울 값을 반환합니다. 컨트롤이 이미 만들어졌으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

7장: Photoshop 통합

Adobe® Dreamweaver CS4®를 통해 Adobe® Photoshop®과 긴밀하게 통합할 수 있습니다. 사용자는 Dreamweaver에서 Photoshop 이미지를 스마트 오브젝트로 삽입할 수 있습니다. Photoshop을 사용하여 원래 이미지를 변경하는 경우 스마트 오브젝트는 Dreamweaver에 있는 이미지를 자동으로 업데이트합니다.

스마트 오브젝트가 동작하는 방법

Photoshop 이미지는 Dreamweaver에서 스마트 오브젝트로 삽입됩니다. 스마트 오브젝트는 원본 Photoshop 이미지에 연결되어 있습니다. 사용자가 Photoshop에서 이미지를 편집하면 Dreamweaver에서 업데이트된 이미지가 표시됩니다. 스마트 오브젝트는 주로 원본 에셋 파일과 웹 이미지의 연결로 인한 특정 상태를 갖습니다. 사용자는 스마트 오브젝트의 상태를 시각적으로 확인할 수 있습니다. 스마트 오브젝트의 동기화 상태는 동기화 표지로 표시됩니다.

스마트 오브젝트 API

스마트 오브젝트 함수는 Dreamweaver 및 Photoshop 통합과 관련된 작업을 처리합니다. 이러한 함수를 사용하여 다음과 같은 작업을 수행할 수 있습니다.

- 이미지 상태 검색
- 이미지의 높이와 폭 검색

dreamweaver.assetPalette.canUpdateSmartObjectFromOriginal()

지원 버전

Dreamweaver CS4

설명

활성자입니다. 이 함수는 "원본에서 업데이트" 명령을 실행할 수 있는 스마트 오브젝트가 [에셋] 패널에서 선택되었는지 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. "원본에서 업데이트"를 현재 선택 영역에 적용할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.assetPalette.updateSmartObjectFromOriginal()

지원 버전

Dreamweaver CS4

설명

이 함수는 연결된 원본 소스 파일의 현재 내용을 기반으로 선택한 웹 이미지를 다시 만듭니다.

인수

없음

반환값

없음

dreamweaver.getSmartObjectState()

지원 버전

Dreamweaver CS4

설명

이 함수는 스마트 오브젝트 기능의 측면에서 웹 이미지의 상태를 반환합니다.

인수

웹 이미지의 절대 로컬 URL입니다.

반환값

다음과 같은 숫자 값으로 스마트 오브젝트의 상태를 반환합니다.

숫자 값	설명
-10	알 수 없는 오류
0	스마트 오브젝트가 없음
1	원본 에셋 파일의 내용과 동기화됨
100	마지막 동기화 후 웹 이미지가 수정됨
200	마지막 동기화 후 원본 에셋이 수정됨
+2	원본 에셋의 크기가 HTML의 폭 및 높이 속성과 다름
+4	웹 이미지의 크기가 HTML의 폭 및 높이 속성과 다름
10	원본 에셋 파일에 액세스할 수 없음
20	웹 이미지 파일에 액세스할 수 없음

dreamweaver.getSmartObjectOriginalWidth()

지원 버전

Dreamweaver CS4

설명

이 함수는 스마트 오브젝트의 원본 에셋 파일에 대한 픽셀 폭을 계산하고 반환합니다.

인수

웹 이미지의 절대 로컬 URL입니다.

반환값

원본 에셋 파일의 픽셀 폭을 반환합니다.

dreamweaver.getImageWidth()

지원 버전

Dreamweaver CS4

설명

이 함수는 이미지의 픽셀 폭을 계산하고 반환합니다.

인수

웹 이미지의 절대 로컬 URL입니다.

반환값

이미지의 픽셀 폭을 반환합니다.

dreamweaver.getImageHeight()

지원 버전

Dreamweaver CS4

설명

이 함수는 이미지의 픽셀 높이를 계산하고 반환합니다.

인수

웹 이미지의 절대 로컬 URL입니다.

반환값

이미지의 픽셀 높이를 반환합니다.

dreamweaver.resolveOriginalAssetFileURLToAbsoluteLocalFilePath())

지원 버전

Dreamweaver CS4

설명

이 함수는 파일 경로를 원본 에셋 파일(디자인 노트에 저장됨)로 확인합니다. 경로는 비어 있거나 사이트 상대적 경로 또는 절대 경로일 수 있습니다.

인수

웹 이미지의 절대 로컬 URL 또는 사이트 상대적 URL입니다. 이 URL은 사이트를 확인하는 데 필요합니다.

반환값

절대 로컬 파일 경로를 반환합니다.

dreamweaver.canUpdateSmartObjectFromOriginal()

지원 버전

Dreamweaver CS4

설명

이 함수는 스마트 오브젝트를 원본 에셋 파일에서 업데이트할 수 있는지 확인합니다.

인수

숫자로 나타낸 스마트 오브젝트의 상태입니다. ImageManipulatorSettings.GetSmartObjectStatus()에서 이 상태를 반환합니다.

반환값

부울 값을 반환합니다. 상태에 따라 원본 이미지에서 업데이트할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.updateSmartObjectFromOriginal()

지원 버전

Dreamweaver CS4

설명

이 함수는 원본 에셋 파일의 현재 내용을 기반으로 웹 이미지를 업데이트합니다.

인수

웹 이미지의 절대 로컬 URL입니다.

반환값

없음

8장: 데이터베이스 API

데이터베이스 API의 함수를 사용하면 데이터베이스 연결을 관리하고 데이터베이스에 저장된 정보에 액세스할 수 있습니다. 데이터베이스 API는 데이터베이스 연결 관리와 데이터베이스 연결 액세스라는 두 가지 용도로 구분됩니다.

데이터베이스 API 함수는 웹 응용 프로그램이 배포되는 런타임이 아니라 사용자가 웹 응용 프로그램을 작성하는 디자인 타임에 사용됩니다.

이러한 함수는 모든 Extension에 사용할 수 있습니다. 실제로 Adobe® Dreamweaver® CS4 서버 비헤이비어, 데이터 서식 및 데이터 소스 API는 모두 이러한 데이터베이스 함수를 사용합니다.

데이터베이스 API 함수 작동 방식

다음 예제에서는 Recordset.js에 대해 서버 비헤이비어 함수 `getDynamicBindings()`를 정의하는 방법을 보여 줍니다. 이 예제에서는 `MMDB.getColumnAndTypeList()` 함수를 사용합니다.

```
function getDynamicBindings(ss)
{
    var serverModel = dw.getDocumentDOM().serverModel.getServerName();
    var bindingsAndTypeArray = new Array();
    var connName=ss.connectionName;
    var statement = ss.source;
    var rsName= ss.rsName;

    // remove SQL comments
    statement = statement.replace(/\/\/*[\S\s]*?\/\//g, " ");
    var bIsSimple = ParseSimpleSQL(statement);
    statement = stripCFIFSimple(statement);

    if (bIsSimple) {
        statement = RemoveWhereClause(statement,false);
    } else {
        var pa = new Array();

        if (ss.ParamArray != null) {
            for (var i = 0; i < ss.ParamArray.length; i++) {
                pa[i] = new Array();
                pa[i][0] = ss.ParamArray[i].name;
                pa[i][1] = ss.ParamArray[i].value;
            }
        }

        var statement = replaceParamsWithVals(statement, pa, serverModel);
    }

    bindingsAndTypeArray = MMDB.getColumnAndTypeList(connName, statement);
    return bindingsAndTypeArray;
}
```

데이터베이스 연결 함수

데이터베이스 연결 함수를 사용하면 Dreamweaver 내장 ADO, ColdFusion 및 JDBC 연결을 비롯한 모든 연결을 만들고 관리할 수 있습니다.

이러한 함수는 연결 관리자에만 연결하여 작동하고 데이터베이스에 액세스하지는 않습니다. 데이터베이스 액세스 함수에 대한 자세한 내용은 65페이지의 “[데이터베이스 액세스 함수](#)”를 참조하십시오.

데이터베이스 연결을 관리하는 경우 사용자 이름과 암호를 가져와서 다음과 같은 작업을 수행할 수 있습니다.

- 데이터베이스에 연결
- [데이터베이스 연결] 대화 상자 열기

MMDB.deleteConnection()

지원 버전

Dreamweaver MX

설명

이 함수는 해당 이름의 데이터베이스 연결을 삭제합니다.

인수

connName

- **connName** 인수는 연결 관리자에 지정된 데이터베이스 연결의 이름입니다. 이 인수는 삭제할 데이터베이스 연결을 이름으로 식별합니다.

반환값

없음

예제

다음 예제에서는 데이터베이스 연결을 삭제합니다.

```
function clickedDelete()  
{  
    var selectedObj = dw.serverComponents.getSelectedNode();  
    if (selectedObj && selectedObj.objectType=="Connection")  
    {  
        var connRec = MMDB.getConnection(selectedObj.name);  
        if (connRec)  
        {  
            MMDB.deleteConnection(selectedObj.name);  
            dw.serverComponents.refresh();  
        }  
    }  
}
```

MMDB.getColdFusionDsnList()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 getRDSUserName() 및 getRDSPassword() 함수를 사용하여 사이트 서버에서 ColdFusion DSN(데이터 소스 이름)을 가져옵니다.

인수
없음

반환값

현재 사이트의 서버에 정의되어 있는 ColdFusion DSN을 포함하는 배열을 반환합니다.

MMDB.getConnection()

지원 버전

Dreamweaver UltraDev 4, Dreamweaver MX에서 기능이 향상됨

설명

이 함수는 해당 이름의 연결 객체를 가져옵니다.

인수

name

- **name** 인수는 참조할 연결의 이름을 지정하는 문자열 변수입니다.

반환값

해당 이름의 연결 객체에 대한 참조를 반환합니다. 연결 객체는 다음 속성을 포함합니다.

속성	설명
name	연결 이름
type	useHTTP가 false일 경우 런타임에 데이터베이스 연결에 사용할 DLL을 나타냅니다.
string	런타임 ADO 연결 문자열 또는 JDBC URL
dsn	ColdFusion DSN
driver	런타임 JDBC 드라이버
username	런타임 사용자 이름
password	런타임 암호
useHTTP	true 또는 false 값을 포함하는 문자열로, 디자인 타임에 원격 드라이버(HTTP 연결)를 사용할지, 아니면 로컬 드라이버(DLL)를 사용할지를 지정합니다.
includePattern	[라이브 데이터] 및 [브라우저에서 미리 보기]를 수행할 때 페이지에서 파일 포함 명령문을 찾는 데 사용된 정규식입니다.
variables	[라이브 데이터]와 [브라우저에서 미리 보기]를 수행할 때 사용되는 페이지 변수 이름 및 해당 값으로 구성된 배열입니다.
catalog	표시되는 메타데이터를 제한하는 데 사용됩니다. 자세한 내용은 68페이지의 " MMDB.getProcedures() "를 참조하십시오.
schema	표시되는 메타데이터를 제한하는 데 사용됩니다. 자세한 내용은 68페이지의 " MMDB.getProcedures() "를 참조하십시오.
filename	연결을 만드는 데 사용된 대화 상자의 파일 이름입니다.

참고: 이러한 속성은 Dreamweaver에서 구현하는 표준 속성입니다. 개발자는 연결 유형을 정의하고 이 표준 집합에 새 속성을 추가하거나 다른 속성 집합을 제공할 수 있습니다.

MMDB.getConnectionList()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 연결 관리자에 정의된 모든 연결 문자열의 목록을 가져옵니다.

인수

없음

반환값

문자열 배열을 반환합니다. 이 경우 각 문자열은 연결 관리자에 나타나는 연결 이름입니다.

예제

MMDB.getConnectionList()를 호출하면 ["EmpDB", "Test", "TestEmp"]와 같은 문자열이 반환될 수 있습니다.

MMDB.getConnectionName()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 연결 문자열에 해당하는 연결 이름을 가져옵니다. 이 함수는 UI(사용자 인터페이스)에서 페이지에 있는 데이터로부터 연결 이름을 다시 선택해야 하는 경우 유용합니다.

두 개의 드라이버를 참조하는 연결 문자열이 있는 경우, 연결 문자열과 반환할 연결 이름에 해당하는 드라이버를 지정할 수 있습니다. 예를 들어, 다음과 같은 두 개의 연결이 있을 수 있습니다.

- 연결 1에는 다음 속성이 있습니다.

```
ConnectionString="jdbc:inetdae:velcro-qa-5:1433?database=pubs"  
DriverName="com.inet.tds.TdsDriver"
```

- 연결 2에는 다음 속성이 있습니다.

```
ConnectionString="jdbc:inetdae:velcro-qa-5:1433?database=pubs"  
DriverName="com.inet.tds.TdsDriver2"
```

연결 1과 연결 2의 연결 문자열은 같습니다. 연결 2는 좀 더 최신 버전의 TdsDriver에 연결되어 있습니다. 드라이버 이름을 이 함수에 전달하여 반환할 연결 이름을 정규화해야 합니다.

인수

connString, {**driverName**}

- **connString** 인수는 연결 이름을 가져올 연결 문자열입니다.
- 선택적인 **driverName** 인수는 **connString** 인수를 더욱 정규화합니다.

반환값

연결 문자열에 해당하는 연결 이름 문자열을 반환합니다.

예제

다음 코드는 문자열 "EmpDB"를 반환합니다.

```
var connectionName = MMDB.getConnectionName (~  
("dsn=EmpDB;uid=;pwd="));
```

MMDB.getConnectionString()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 해당 이름의 연결과 연관된 연결 문자열을 가져옵니다.

인수

connName

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.

반환값

해당 이름의 연결에 해당하는 연결 문자열을 반환합니다.

예제

var connectionString = MMDB.getConnectionString("EmpDB") 코드는 ADO 또는 JDBC 연결에 대해 서로 다른 문자열을 반환합니다.

- ADO 연결의 경우 다음 문자열이 반환될 수 있습니다.

```
"dsn=EmpDB;uid=;pwd=";
```

- JDBC 연결의 경우 다음 문자열이 반환될 수 있습니다.

```
"jdbc:inetdae:192.168.64.49:1433?database=pubs&user=JoeUser&  
password=joesSecret"
```

MMDB.getDriverName()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 연결과 연관된 드라이버 이름을 가져옵니다. JDBC 연결의 경우에만 드라이버 이름이 있습니다.

인수

connName

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.

반환값

드라이버 이름을 포함하는 문자열을 반환합니다.

예제

MMDB.getDriverName("EmpDB"); 명령문은 다음 문자열을 반환할 수 있습니다.

```
"jdbc/oracle/driver/JdbcOracle"
```

MMDB.getLocalDsnList()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 사용자 시스템에 정의된 ODBC DSN을 가져옵니다.

인수

없음

반환값

사용자 시스템에 정의된 ODBC DSN을 포함하는 배열을 반환합니다.

MMDB.getPassword()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 연결에 사용되는 암호를 가져옵니다.

인수

connName

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.

반환값

연결 이름과 연관된 암호 문자열을 반환합니다.

예제

MMDB.getPassword("EmpDB"); 명령문은 "joessecret"를 반환할 수 있습니다.

MMDB.getRDSPassword()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 RDS(Remote Development Services) 암호(ColdFusion 연결에 사용)를 가져옵니다.

인수

없음

반환값

RDS 암호를 포함하는 문자열을 반환합니다.

MMDB.getRDSUserName()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 RDS 사용자 이름(ColdFusion 연결에 사용)을 가져옵니다.

인수

없음

반환값

RDS 사용자 이름을 포함하는 문자열을 반환합니다.

MMDB.getRemoteDsnList()

지원 버전

Dreamweaver UltraDev 4, Dreamweaver MX에서 기능이 향상됨

설명

이 함수는 사이트 서버에서 ODBC DSN을 가져옵니다. 현재 사이트의 서버 모델이 ColdFusion인 경우 `getRDSUserName()` 및 `getRDSPassword()` 함수가 사용됩니다. 이 함수를 사용하여 개발자는 `MMDB.getRemoteDsnList()`를 통해 생성된 원격 연결 URL에 추가할 URL 매개 변수 문자열을 지정할 수 있습니다. 개발자가 매개 변수 문자열을 제공하면 이 함수는 해당 매개 변수 문자열을 HTTP 연결 스크립트에 전달합니다.

인수

{urlParams}

- 선택적인 **urlParams** 인수는 앰퍼샌드(&) 문자로 구분된 **name=value** 표현식 목록을 포함하는 문자열입니다. 이 경우 값을 따옴표로 묶으면 안 됩니다. 일부 문자(예: 값 Hello World에 포함된 공백)는 인코딩이 필요합니다. `a=1&b=Hello%20World` 예제에서는 `MMDB.getRemoteDsnList()`에 전달할 수 있는 유효한 인수의 예를 보여 줍니다.

반환값

현재 사이트의 서버에 정의된 ODBC DSN을 포함하는 배열을 반환합니다.

MMDB.getRuntimeConnectionType()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 연결 이름의 런타임 연결 유형을 반환합니다.

인수**connName**

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.

반환값

연결 유형에 해당하는 문자열을 반환합니다. 이 함수는 "ADO", "ADODSN", "JDBC" 또는 "CFDSN" 값 중 하나를 반환할 수 있습니다.

예제

다음 코드는 ADO 연결에 대한 문자열 "ADO"를 반환합니다.

```
var connectionType = MMDB.getRuntimeConnectionType ("EmpDB")
```

MMDB.getUserName()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 연결에 대한 사용자 이름을 반환합니다.

인수**connName**

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.

반환값

연결 이름과 연관된 사용자 이름 문자열을 반환합니다.

예제

MMDB.getUserName ("EmpDB"); 명령문은 "amit"를 반환할 수 있습니다.

MMDB.hasConnectionWithName()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 지정된 이름의 연결이 존재하는지 여부를 확인합니다.

인수**name**

- **name** 인수는 연결 이름입니다.

반환값

부울 값을 반환합니다. 지정된 이름의 연결이 존재하면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

MMDB.needToPromptForRdsInfo()

지원 버전

Dreamweaver MX

설명

이 함수는 Dreamweaver에서 [RDS 로그인 정보] 대화 상자를 열지 여부를 확인합니다.

인수**bForce**

- **bForce** 인수는 부울 값입니다. **true**로 설정하면 이전에 RDS 로그인 대화 상자를 취소했어도 RDS 로그인 정보를 묻는 메시지가 표시됩니다.

반환값

부울 값을 반환합니다. RDS 로그인 정보를 묻는 메시지가 표시되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

MMDB.needToRefreshColdFusionDsnList()

지원 버전

Dreamweaver MX

설명

이 함수는 사용자가 다음 번에 목록을 요청할 경우 캐시를 비우고 응용 프로그램 서버에서 ColdFusion 데이터 소스 목록을 가져오도록 연결 관리자에 지시합니다.

인수

없음

반환값

없음

MMDB.popupConnection()

지원 버전

Dreamweaver MX

설명

이 함수는 연결 대화 상자를 시작합니다. 이 함수는 다음과 같은 세 가지 방법으로 사용됩니다.

- 인수 목록이 **dialogFileName**(문자열)으로만 구성된 경우 **popupConnection()** 함수를 실행하면 Dreamweaver에서 새 연결을 정의할 수 있도록 [연결] 대화 상자가 열립니다.
- 인수 목록이 **connRec**(연결 참조)로만 구성된 경우 **popupConnection()** 함수를 실행하면 Dreamweaver에서 해당 이름의 연결을 편집할 수 있도록 [연결] 대화 상자가 편집 모드로 실행됩니다. 이 모드에서는 이름 텍스트 필드가 흐리게 나타납니다.
- 인수 목록이 **connRec** 및 부울 값 **bDuplicate**로 구성된 경우 **popupConnection()** 함수를 실행하면 Dreamweaver에서 [연결] 대화 상자가 복제 모드로 열립니다. 이 모드에서는 이름 텍스트 필드가 빈 상태가 되며 나머지 속성은 복사되어 복제 연결이 정의됩니다.

인수

dialogFileName 또는 **connRec** 또는 **connrec**, **bDuplicate**

- dialogFileName** 인수는 Configuration/Connections/서버 모델 폴더에 있는 HTML 파일의 이름을 포함하는 문자열입니다. 이 HTML 파일은 연결을 만드는 대화 상자를 정의합니다. 이 파일은 **findConnection()**, **inspectConnection()** 및 **applyConnection()** 함수를 구현해야 합니다. 일반적으로 이러한 함수를 구현하는 JavaScript 파일을 하나 만든 다음 HTML 파일에 포함합니다. 연결 만들기에 대한 자세한 내용은 76페이지의 “[데이터베이스 연결 API](#)”를 참조하십시오.
- connRec** 인수는 기존 Connection 객체에 대한 참조입니다.
- bDuplicate** 인수는 부울 값입니다.

반환값

없음. 정의된 연결 대화 상자가 나타납니다.

MMDB.setRDSPassword()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 RDS 암호를 설정합니다.

인수

password

- password** 인수는 RDS 암호를 포함하는 문자열입니다.

반환값

없음

MMDB.setRDSUserName()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 RDS 사용자 이름을 설정합니다.

인수

username

- **username** 인수는 유효한 RDS 사용자 이름입니다.

반환값

없음

MMDB.showColdFusionAdmin()

지원 버전

Dreamweaver MX

설명

이 함수는 [ColdFusion Administrator] 대화 상자를 표시합니다.

인수

없음

반환값

없음. [ColdFusion Administrator] 대화 상자가 나타납니다.

MMDB.showConnectionMgrDialog()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 [연결 관리자] 대화 상자를 표시합니다.

인수

없음

반환값

없음. [연결 관리자] 대화 상자가 나타납니다.

MMDB.showOdbcDialog()

지원 버전

Dreamweaver UltraDev 4(Windows 전용)

설명

이 함수는 [시스템 ODBC 관리] 대화 상자나 [ODBC 데이터 원본 관리자] 대화 상자를 표시합니다.

인수

없음

반환값

없음. [시스템 ODBC 관리] 대화 상자나 [ODBC 데이터 원본 관리자] 대화 상자가 나타납니다.

MMDB.showRdsUserDialog()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 [RDS 사용자 이름 및 암호] 대화 상자를 표시합니다.

인수

username, password

- **username** 인수는 사용자 이름의 초기값입니다.
- **password** 인수는 암호의 초기값입니다.

반환값

username과 password 속성에 새 값이 포함된 객체를 반환합니다. 속성이 정의되지 않았으면 사용자가 대화 상자를 취소한 것입니다.

MMDB.showRestrictDialog()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 [제한] 대화 상자를 표시합니다.

인수

catalog, schema

- **catalog** 인수는 카탈로그의 초기값입니다.
- **schema** 인수는 스키마의 초기값입니다.

반환값

catalog 및 schema 속성에 새 값이 포함된 객체를 반환합니다. 속성이 정의되지 않았으면 사용자가 대화 상자를 취소한 것입니다.

MMDB.testConnection()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 연결 설정을 테스트하고 결과를 나타내는 모달 대화 상자를 표시합니다.

인수

serverPropertiesArray

이 함수에는 하나의 인수(다음 목록 중 현재 서버 모델에 적합한 값이 포함된 배열 객체)가 사용됩니다. 테스트할 연결에 적용되지 않는 속성은 빈 문자열("")로 설정해야 합니다.

- **type** 인수는 **useHTTP** 값이 **false**인 경우 연결 설정을 테스트하기 위해 디자인 타임에 데이터베이스에 연결하는 데 사용할 DLL을 나타냅니다.
- **string** 인수는 ADO 연결 문자열 또는 JDBC URL입니다.
- **dsn** 인수는 데이터 소스 이름입니다.
- **driver** 인수는 JDBC 드라이버입니다.
- **username** 인수는 사용자 이름입니다.
- **password** 인수는 암호입니다.
- **useHTTP** 인수는 부울 값입니다. **true**로 설정하면 디자인 타임에 Dreamweaver에서 HTTP 연결을 사용하고 그렇지 않으면 Dreamweaver에서 DLL을 사용합니다.

반환값

부울 값을 반환합니다. 연결 테스트가 성공적이면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

데이터베이스 액세스 함수

데이터베이스 액세스 함수를 사용하면 데이터베이스 쿼리를 수행할 수 있습니다.

데이터베이스 정보에 액세스하는 경우, 데이터베이스의 스키마 또는 구조를 나타내는 메타데이터를 가져오는 등의 작업을 수행할 수 있습니다. 메타데이터에는 테이블, 열, 저장 프로시저 및 뷰의 이름과 같은 정보가 포함됩니다. 또한 데이터베이스 쿼리 또는 저장 프로시저를 실행한 결과를 표시할 수도 있습니다. 이 API를 통해 데이터베이스에 액세스하는 경우 SQL(structured query language) 문을 사용합니다.

데이터베이스 연결을 관리하는 함수 컬렉션은 53페이지의 “[데이터베이스 연결 함수](#)”를 참조하십시오.

다음 목록에서는 사용 가능한 함수에 공통으로 사용되는 몇 가지 인수에 대해 설명합니다.

- 대부분의 데이터베이스 액세스 함수는 연결 이름을 인수로 사용합니다. 연결 관리자에서 유효한 연결 이름의 목록을 볼 수도 있고 `MMDB.getConnectionList()` 함수를 사용하여 프로그래밍 방식으로 연결 이름의 전체 목록을 가져올 수도 있습니다.
- 경우에 따라 저장 프로시저에도 매개 변수가 필요할 때가 있습니다. 두 가지 방식으로 데이터베이스 액세스 함수에 매개 변수 값을 지정할 수 있습니다. 첫째, 매개 변수 값의 배열(`paramValuesArray`)을 제공할 수 있습니다. 매개 변수 값만 지정하는 경우에는 값의 순서가 저장 프로시저에 필요한 매개 변수의 순서와 같아야 합니다. 둘째, 매개 변수 이름의 배열(`paramNameArray`)을 제공하는 매개 변수 값을 지정합니다. `MMDB.getSPPParamsAsString()` 함수를 사용하여 저장 프로시저의 매개 변수를 가져올 수 있습니다. 매개 변수 이름을 제공하는 경우 `paramValuesArray`에서 지정하는 값은 `paramNameArray`에서 매개 변수 이름을 지정하는 순서와 같아야 합니다.

MMDB.getColumnAndTypeList()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 실행된 SQL SELECT 문에서 열과 해당 유형의 목록을 가져옵니다.

인수**connName, statement**

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **statement** 인수는 실행할 SQL SELECT 문입니다.

반환값

지정된 SELECT 문과 일치하는 열(및 해당 유형)의 목록을 나타내는 문자열 배열을 반환하거나, SQL 문이 잘못되었거나 연결할 수 없는 경우에는 오류를 반환합니다.

예제

var columnArray = MMDB.getColumnAndTypeList("EmpDB","Select * from Employees") 코드는 다음 문자열 배열을 반환합니다.

```
columnArray[0] = "EmpName"
columnArray[1] = "varchar"
columnArray[2] = "EmpFirstName"
columnArray[3] = "varchar"
columnArray[4] = "Age"
columnArray[5] = "integer"
```

MMDB.getColumnList()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 실행된 SQL SELECT 문에서 열 목록을 가져옵니다.

인수**connName, statement**

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **statement** 인수는 실행할 SQL SELECT 문입니다.

반환값

SELECT 문과 일치하는 열 목록을 나타내는 문자열 배열을 반환하거나, SQL 문이 잘못되었거나 연결할 수 없는 경우에는 오류를 반환합니다.

예제

var columnArray = MMDB.getColumnList("EmpDB","Select * from Employees") 코드는 다음 문자열 배열을 반환합니다.

```
columnArray[0] = "EmpName"
columnArray[1] = "EmpFirstName"
columnArray[2] = "Age"
```

MMDB.getColumnns()

지원 버전

Dreamweaver MX, Dreamweaver MX 2004에서 인수가 업데이트됨

설명

이 함수는 지정된 테이블의 열을 설명하는 객체 배열을 반환합니다.

인수

connName, tableName

- **connName** 인수는 연결 이름입니다. 이 값은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 문자열이 들어 있는 연결을 식별합니다.
- **tableName** 인수는 쿼리할 테이블입니다.

반환값

각 객체가 하나의 열을 나타내는 객체 배열을 반환합니다. 각 객체는 해당 열과 연관된 다음 세 가지 속성을 정의합니다.

속성 이름	설명
name	열 이름(예: price)
datatype	열의 데이터 유형(예: small money)
definedsize	정의된 열 크기(예: 8)
nullable	열이 null 값을 포함할 수 있는지 여부를 나타냅니다.

예제

다음 예제에서는 MMDB.getColumnns()를 사용하여 도구 설명 텍스트 값을 설정합니다.

```
var columnNameObjs = MMDB.getColumnns(connName, tableName);
var databaseType = MMDB.getDatabaseType(connName);
for (i = 0; i < columnNameObjs.length; i++)
{
    var columnObj = columnNameObjs[i];
    var columnName = columnObj.name;
    var typename = columnObj.datatype;
    if (dwscripts.isNumber(typename))
    {
        // it already is a num
        typename = dwscripts.getDBColumnTypeAsString(typename, databaseType);
    }
    var tooltipText = typename;
}
```

MMDB.getColumnsOfTable()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 테이블에 있는 모든 열의 목록을 가져옵니다.

인수**connName, tableName**

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **tableName** 인수는 **connName** 인수로 지정된 데이터베이스의 테이블 이름입니다.

반환값

문자열 배열을 반환합니다. 이 경우 각 문자열은 테이블에 있는 열 이름을 나타냅니다.

예제

MMDB.getColumnsOfTable ("EmpDB","Employees"); 명령문은 다음 문자열을 반환합니다.

["EmpID", "FirstName", "LastName"]

MMDB.getPrimaryKeys()

지원 버전

Dreamweaver MX

설명

이 함수는 해당 이름의 테이블에 대한 기본 키를 구성하기 위해 결합되는 열 이름을 반환합니다. 기본 키는 데이터베이스 행의 고유 식별자이며 하나 이상의 열로 구성됩니다.

인수**connName, tableName**

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **tableName** 인수는 해당 테이블의 기본 키를 구성하는 열 집합을 검색할 테이블의 이름입니다.

반환값

문자열 배열을 반환합니다. 이 배열에는 기본 키를 구성하는 각 열당 하나의 문자열이 포함됩니다.

예제

다음 예제에서는 지정된 테이블에 대한 기본 키를 반환합니다.

```
var connName      = componentRec.parent.parent.name;
var tableName     = componentRec.name;
var primaryKeys   = MMDB.getPrimaryKeys(connName,tableName);
```

MMDB.getProcedures()

지원 버전

Dreamweaver MX

설명

이 함수는 해당 이름의 연결과 연관된 프로시저 객체의 배열을 반환합니다.

인수

connName

- connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.

반환값

프로시저 객체의 배열을 반환합니다. 각 프로시저 객체에는 다음 세 가지 속성이 있습니다.

속성 이름	설명
schema	객체와 연관된 스키마의 이름입니다. 이 속성은 getProcedures() 함수가 액세스하는 SQL 데이터베이스의 저장 프로시저와 연관된 사용자를 식별합니다. 이 함수가 액세스하는 데이터베이스는 연결 유형에 따라 달라집니다. - ODBC 연결의 경우 ODBC 데이터 원본이 데이터베이스를 정의합니다. DSN은 getProcedures() 함수에 전달되는 연결 객체(connName)의 dsn 속성으로 지정됩니다. - OLE DB 연결의 경우 연결 문자열에 데이터베이스 이름이 지정됩니다.
catalog	객체(소유자 한정자)와 연관된 카탈로그의 이름입니다. catalog 속성의 값은 OLE DB 드라이버의 속성으로 정의됩니다. 이 드라이버 속성은 OLE DB 연결 문자열에 데이터베이스가 지정되지 않은 경우 사용할 기본 user.database를 정의합니다.
procedure	프로시저 이름입니다.

참고: Dreamweaver는 사용자가 레코드셋을 수정할 때마다 데이터베이스에 연결하여 모든 테이블을 가져옵니다. 데이터베이스에 테이블이 너무 많은 경우 시스템에 따라 Dreamweaver에서 이러한 테이블을 검색하는 데 오랜 시간이 걸릴 수 있습니다. 데이터베이스에 스키마나 카탈로그가 포함된 경우에는 디자인 타임에 이 스키마나 카탈로그를 사용하여 Dreamweaver가 가져올 데이터 항목 수를 제한할 수 있습니다. 이를 Dreamweaver에서 적용하려면 먼저 데이터베이스 응용 프로그램에서 스키마나 카탈로그를 만들어야 합니다. 자세한 내용은 데이터베이스 설명서를 참조하거나 시스템 관리자에게 문의하십시오.

예제

다음 코드에서는 프로시저 목록을 가져옵니다.

```
var procObjects = MMDB.getProcedures(connectionName);
for (i = 0; i < procObjects.length; i++)
{
    var thisProcedure = procObjects[i]
    thisSchema = Trim(thisProcedure.schema)
    if (thisSchema.length == 0)
    {
        thisSchema = Trim(thisProcedure.catalog)
    }
    if (thisSchema.length > 0)
    {
        thisSchema += "."
    }

    var procName = String(thisSchema + thisProcedure.procedure);
}
```

MMDB.getSPColumnList()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 저장 프로시저를 호출하여 생성된 결과 집합 열의 목록을 가져옵니다.

인수

connName, statement, paramValuesArray

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **statement** 인수는 실행 시 결과 집합을 반환하는 저장 프로시저의 이름입니다.
- **paramValuesArray** 인수는 디자인 타임에 지정한 매개 변수 테스트 값 목록을 포함하는 배열입니다. 저장 프로시저에 필요한 순서대로 매개 변수 값을 지정해야 합니다. MMDB.getSPParamsAsString() 함수를 사용하여 저장 프로시저의 매개 변수를 가져올 수 있습니다.

반환값

열 목록을 나타내는 문자열 배열을 반환합니다. 이 함수는 SQL 문 또는 연결 문자열이 잘못된 경우 오류를 반환합니다.

예제

다음 코드는 getNewEmployeesMakingAtLeast 저장 프로시저를 실행하여 생성된 결과 집합 열의 목록을 반환할 수 있습니다.

```
var paramValueArray = new Array("2/1/2000", "50000")
var columnArray = MMDB.getSPColumnList("EmpDB", "\n" +
    "getNewEmployeesMakingAtLeast", paramValueArray)
The following values return:
columnArray[0] = "EmpID", columnArray[1] = "LastName", "\n" +
columnArray[2] = "startDate", columnArray[3] = "salary"
```

MMDB.getSPColumnListNamedParams()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 저장 프로시저를 호출하여 생성된 결과 집합 열의 목록을 가져옵니다.

인수

connName, statement, paramNameArray, paramValuesArray

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **statement** 인수는 실행 시 결과 집합을 반환하는 저장 프로시저의 이름입니다.
- **paramNameArray** 인수는 매개 변수 이름 목록을 포함하는 배열입니다. MMDB.getSPParamsAsString() 함수를 사용하여 저장 프로시저의 매개 변수를 가져올 수 있습니다.
- **paramValuesArray** 인수는 디자인 타임에 지정한 매개 변수 테스트 값 목록을 포함하는 배열입니다. 해당 프로시저가 실행될 때 매개 변수를 필요로 하는지 여부를 지정할 수 있습니다. 매개 변수 이름을 **paramNameArray**에 지정한 경우 **paramNameArray**에 나타나는 해당 매개 변수 이름 순서와 같은 순서로 매개 변수 값을 지정해야 합니다. **paramNameArray**를 제공하지 않은 경우에는 저장 프로시저에 필요한 순서대로 값을 지정해야 합니다.

반환값

열 목록을 나타내는 문자열 배열을 반환합니다. 이 함수는 SQL 문 또는 연결 문자열이 잘못된 경우 오류를 반환합니다.

예제

다음 코드는 `getNewEmployeesMakingAtLeast` 저장 프로시저를 실행하여 생성된 결과 집합 열의 목록을 반환할 수 있습니다.

```
var paramNameArray = new Array("startDate", "salary")
var paramValueArray = new Array("2/1/2000", "50000")
var columnArray = MMDB.getSPColumnListNamedParams("EmpDB", ~
"getNewEmployeesMakingAtLeast", paramNameArray, paramValueArray)
```

다음 값이 반환됩니다.

```
columnArray[0] = "EmpID", columnArray[1] = "LastName", ~
columnArray[2] = "startDate", columnArray[3] = "salary"
```

MMDB.getSPParameters()

지원 버전

Dreamweaver MX

설명

이 함수는 해당 이름의 프로시저에 대한 매개 변수 객체 배열을 반환합니다.

인수

connName, procName

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **procName** 인수는 프로시저 이름입니다.

반환값

매개 변수 객체의 배열을 반환합니다. 각 객체는 다음과 같은 속성 집합을 지정합니다.

속성 이름	설명
name	매개 변수 이름(예: @lolimit)
datatype	매개 변수의 데이터 유형(예: smallmoney)
direction	매개 변수의 전달 방향 1- 매개 변수가 입력 전용으로 사용됩니다. 2- 매개 변수가 출력 전용으로 사용됩니다. 이 경우 매개 변수를 참조로(by reference) 전달하면 메서드가 해당 매개 변수에 값을 넣습니다. 그러면 메서드가 반환된 후 해당 값을 사용할 수 있습니다. 3- 매개 변수가 입/출력에 사용됩니다. 4- 매개 변수에 반환값이 저장됩니다.

예제

다음 예제에서는 지정된 프로시저에 대한 매개 변수 객체를 검색하고 해당 속성을 사용하여 각 객체에 대한 도구 설명을 만듭니다.

```
var paramNameObjs = MMDB.getSPPParameters(connName,procName);
for (i = 0; i < paramNameObjs.length; i++)
{
    var paramObj = paramNameObjs[i];
    var tooltipText = paramObj.datatype;
    tooltipText+=" ";
    tooltipText+=GetDirString(paramObj.directiontype);
}
```

MMDB.getSPPParamsAsString()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 저장 프로시저가 사용하는 매개 변수 목록을 포함하는 문자열을 가져옵니다. 각 매개 변수는 쉼표로 구분됩니다.

인수

connName, procName

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **procName** 인수는 저장 프로시저 이름입니다.

반환값

저장 프로시저에 필요한 매개 변수 목록을 포함하는 문자열을 반환합니다. 각 매개 변수는 쉼표로 구분됩니다. 매개 변수 이름, 방향 및 데이터 유형은 세미콜론(;)으로 구분되어 포함됩니다.

예제

MMDB.getSPPParamsAsString("EmpDB","getNewEmployeesMakingAtLeast") 코드는 startDate;direction:in;datatype:date,salary;direction:in;datatype:integer와 같은 형태의 문자열을 반환할 수 있습니다.

이 예제에서 저장 프로시저 getNewEmployeesMakingAtLeast에는 startDate 및 Salary라는 두 매개 변수가 있습니다. startDate의 경우 방향은 in이고 데이터 유형은 date입니다. salary는 방향이 in이고 데이터 유형은 date입니다.

MMDB.getTables()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 데이터베이스에 정의된 모든 테이블의 목록을 가져옵니다. 각 테이블 객체에는 table, schema 및 catalog 속성이 있습니다.

인수

connName

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.

반환값

객체의 배열을 반환합니다. 각 객체에는 `table`, `schema` 및 `catalog` 속성이 있습니다. `Table`은 테이블 이름입니다. `Schema`는 테이블을 포함하는 스키마 이름입니다. `Catalog`는 테이블을 포함하는 카탈로그입니다.

예제

`MMDB.getTables("EmpDB");` 문은 두 개의 객체로 구성된 배열을 만들 수 있습니다. 첫 번째 객체의 속성은 다음 예제와 유사합니다.

```
object1[table:"Employees", schema:"personnel", catalog:"syscat"]
```

두 번째 객체의 속성은 다음 예제와 유사합니다.

```
object2[table:"Departments", schema:"demo", catalog:"syscat2"]
```

MMDB.getViews()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 지정된 데이터베이스에 정의된 모든 뷰의 목록을 가져옵니다. 각 뷰 객체에는 `catalog`, `schema` 및 `view` 속성이 있습니다.

인수**connName**

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.

반환값

뷰 객체의 배열을 반환합니다. 각 객체에는 `catalog`, `schema` 및 `view` 속성이 있습니다. `catalog` 또는 `schema`는 연결 정보의 일부로 정의된 각 스키마 이름이나 카탈로그 이름과 관련이 있는 뷰의 수를 제한하거나 필터링하는 데 사용됩니다.

예제

다음 예제는 주어진 연결 값 `CONN_LIST.getValue()`에 대한 뷰를 반환합니다.

```
var viewObjects = MMDB.getViews(CONN_LIST.getValue())
for (i = 0; i < viewObjects.length; i++)
{
    thisView = viewObjects[i]
    thisSchema = Trim(thisView.schema)
    if (thisSchema.length == 0)
    {
        thisSchema = Trim(thisView.catalog)
    }
    if (thisSchema.length > 0)
    {
        thisSchema += "."
    }
    views.push(String(thisSchema + thisView.view))
}
```

MMDB.showResultset()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 SQL 문을 실행한 결과를 포함하는 대화 상자를 표시합니다. 대화 상자는 표 형식으로 표시되며 머리글에는 결과 집합을 설명하는 열 정보가 제공됩니다. 연결 문자열이나 SQL 문이 잘못된 경우 오류 메시지가 나타납니다. 이 함수는 SQL 문의 유효성을 검사합니다.

인수

connName, SQLstatement

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **SQLstatement** 인수는 SQL SELECT 문입니다.

반환값

없음. 이 함수는 SQL 문 또는 연결 문자열이 잘못된 경우 오류를 반환합니다.

예제

다음 코드에서는 실행된 SQL 문의 결과를 표시합니다.

```
MMDB.showResultset("EmpDB", "Select EmpName, EmpFirstName, Age  
from Employees")
```

MMDB.showSPResultset()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 저장 프로시저를 실행한 결과가 포함된 대화 상자를 표시합니다. 대화 상자는 표 형식으로 표시되며 머리글에는 결과 집합을 설명하는 열 정보가 제공됩니다. 연결 문자열이나 저장 프로시저가 잘못된 경우 오류 메시지가 나타납니다. 이 함수는 저장 프로시저의 유효성을 검사합니다.

인수

connName, procName, paramValuesArray

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **procName** 인수는 실행할 저장 프로시저 이름입니다.
- **paramValuesArray** 인수는 디자인 타임에 지정한 매개 변수 테스트 값 목록을 포함하는 배열입니다. 저장 프로시저에 필요한 순서대로 매개 변수 값을 지정해야 합니다. MMDB.getSPParamsAsString() 함수를 사용하여 저장 프로시저의 매개 변수를 가져올 수 있습니다.

반환값

이 함수는 SQL 문 또는 연결 문자열이 잘못되었으면 오류를 반환하고 그렇지 않으면 아무 것도 반환하지 않습니다.

예제

다음 코드에서는 실행된 저장 프로시저의 결과를 표시합니다.

```
var paramValueArray = new Array("2/1/2000", "50000")
MMDB.showSPResultset("EmpDB", "getNewEmployeesMakingAtLeast", ~
paramValueArray)
```

MMDB.showSPResultsetNamedParams()

지원 버전

Dreamweaver UltraDev 1

설명

이 함수는 지정된 저장 프로시저의 결과 집합이 포함된 대화 상자를 표시합니다. 대화 상자는 표 형식으로 표시되며 머리글에는 결과 집합을 설명하는 열 정보가 제공됩니다. 연결 문자열이나 저장 프로시저가 잘못된 경우 오류 메시지가 나타납니다. 이 함수는 저장 프로시저의 유효성을 검사합니다. 이 함수는 매개 변수 값을 저장 프로시저에 필요한 순서대로 지정할 수 없고 이름별로 지정할 수 있기 때문에 MMDB.showSPResultset() 함수와 다릅니다.

인수

connName, procName, paramNameArray, paramValuesArray

- **connName** 인수는 연결 관리자에 지정된 연결 이름입니다. 이 이름은 Dreamweaver에서 라이브 데이터 소스에 대한 데이터베이스 연결을 만들 때 사용하는 연결 문자열을 식별합니다.
- **procName** 인수는 실행될 경우 결과 집합을 반환하는 저장 프로시저의 이름입니다.
- **paramNameArray** 인수는 매개 변수 이름 목록을 포함하는 배열입니다. MMDB.getSPParamsAsString() 함수를 사용하여 저장 프로시저의 매개 변수를 가져올 수 있습니다.
- **paramValuesArray** 인수는 디자인 타임에 지정한 매개 변수 테스트 값 목록을 포함하는 배열입니다.

반환값

이 함수는 SQL 문 또는 연결 문자열이 잘못되었으면 오류를 반환하고 그렇지 않으면 아무 것도 반환하지 않습니다.

예제

다음 코드에서는 실행된 저장 프로시저의 결과를 표시합니다.

```
var paramNameArray = new Array("startDate", "salary")
var paramValueArray = new Array("2/1/2000", "50000")
MMDB.showSPResultsetNamedParams("EmpDB", "getNewEmployees~
MakingAtLeast", paramNameArray, paramValueArray)
```

9장: 데이터베이스 연결 API

개발자는 Adobe® Dreamweaver®의 새 서버 모델 또는 기존 서버 모델에 대해 새로운 연결 유형과 해당 대화 상자를 만들 수 있습니다. 이 경우 사용자가 사이트를 설정하여 페이지 작성을 시작할 때 연결 객체를 만들 수 있습니다. 연결 객체를 만들려면 사용자가 먼저 개발자가 만든 특정 연결 유형을 선택해야 합니다.

새 연결 유형 선택

다음과 같은 방법으로 새로운 연결 유형을 선택할 수 있습니다.

- [응용 프로그램] 패널에서 플러스(+) 버튼을 클릭하고 [레코드세트]를 선택할 수 있습니다. [레코드세트] 대화 상자에서 [연결] 팝업 메뉴를 확장할 수 있습니다.
- [데이터베이스] 패널의 [데이터베이스] 탭에서 플러스(+) 버튼을 클릭하고 [데이터 소스 이름]을 선택할 수 있습니다.

새 연결 유형 개발

다음 단계에서는 새 연결 유형을 만드는 절차를 간략하게 설명합니다.

1 연결 대화 상자의 레이아웃을 만듭니다.

연결 대화 상자의 UI(사용자 인터페이스)에 대한 레이아웃을 지정하는 HTML 파일을 만듭니다. 연결 이름을 사용하여 이 파일의 이름을 지정합니다(예: myConnection.htm). 대화 상자 만들기에 대한 자세한 내용은 **Dreamweaver** 시작하기를 참조하십시오.

다음 예제에 표시된 대로 2단계, "최소한 다음 요소를 구현하는 JavaScript 파일을 만듭니다."에서 정의하는 JavaScript 구현 파일에 대한 참조가 이 HTML 파일에 포함되어야 합니다.

```
<head>
  <script SRC="../../myConnectionImpl.js"></script>
</head>
```

연결 대화 상자를 정의하는 이 HTML 파일을 Configuration/Connections/server-model/platform 폴더에 저장합니다. 여기서 platform은 Windows 또는 Macintosh입니다.

예를 들어, Windows 플랫폼의 ASP JavaScript 문서에 대한 기본 ADO 연결 대화 상자는 ASP_Js/Win 폴더에 저장되며 이름은 Connection_ado_conn_string.htm입니다.

참고: 런타임에 Dreamweaver에서는 ASP_Js/Win 폴더에 있는 대화 상자 모음에서 사용자가 사용할 수 있는 연결 유형 목록을 동적으로 구성합니다.

Configuration/ServerModels 폴더에는 각 서버 모델을 정의하는 HTML 파일이 있습니다. 각 HTML 파일에는 해당 서버 모델과 연관된 폴더의 이름을 반환하는 getServerModelFolderName() 함수가 있습니다. 다음 예제에서는 ASP JavaScript 문서 형식에 대한 함수를 보여 줍니다.

```
function getServerModelFolderName()
{
    return "ASP_JS";
}
```

Configuration/DocumentTypes 폴더에 있는 MMDocumentTypes.xml 파일을 통해 서버 모델과 문서 형식 간의 매핑을 확인할 수도 있습니다.

2 최소한 다음 요소를 구현하는 JavaScript 파일을 만듭니다.

요소	설명	예제
변수 집합	각 변수는 특정 연결 속성을 정의합니다.	연결 유형, 데이터 소스 이름 등
버튼 집합	각 버튼이 연결 대화 상자에 나타납니다.	[테스트], [도움말] 등([확인] 및 [취소]는 자동으로 포함됨)
연결 함수	이러한 함수가 함께 사용되어 연결 API를 정의합니다.	• findConnection() • applyConnection() • inspectConnection()

이 구현 파일의 이름은 임의로 선택할 수 있지만 확장명은 .js를 사용해야 합니다(예: myConnectionImpl.js). 이 구현 파일은 로컬 컴퓨터나 원격 컴퓨터에 저장할 수 있으며, Configuration/Connections 폴더의 해당 하위 폴더에 저장할 수도 있습니다.

참고: 1단계, "연결 대화 상자의 레이아웃을 만듭니다."에서 정의한 HTML 파일에는 이 연결 유형 구현 파일이 포함되어야 합니다.

표준 connection_includefile.edml 파일에 제공된 매개 변수 이외에 다른 연결 매개 변수를 정의할 필요가 없는 경우에는 이 두 단계만 사용해도 새 연결 대화 상자를 만들 수 있습니다.

참고: 사용자에게 표시되는 대화 상자의 제목은 HTML 문서에서 **title** 태그에 지정됩니다.

다음 단원에 나열된 함수를 사용하면 연결 대화 상자를 만들 수 있습니다. 사용자를 위해 포함 파일을 생성하는 호출을 구현하면서 연결 XML 파일의 서버 모델 섹션 내에 연결 유형을 등록할 수 있습니다.

새 연결 만들기 및 연관된 데이터베이스 연결 API에 대한 자세한 내용은 53페이지의 “데이터베이스 연결 함수”를 참조하십시오.

연결 API

사용자에게 표시되는 대화 상자를 포함하여 새로운 연결 유형을 만들려면 findConnection(), inspectConnection() 및 applyConnection() 함수를 구현해야 합니다. 이 세 함수를 작성하여 새 연결 유형과 연관된 JavaScript 구현 파일에 포함시킵니다(2단계, “최소한 다음 요소를 구현하는 JavaScript 파일을 만듭니다.”를 참조).

applyConnection() 함수는 포함 파일 내의 HTML 소스를 반환합니다. HTML 소스의 예제는 80페이지의 “생성된 포함 파일”에 있습니다. findConnection() 함수는 HTML 소스를 가져와 해당 속성을 추출합니다. XML 파일에서 검색 패턴을 사용하여 applyConnection()에서 반환되는 정보를 추출하도록 findConnection()을 구현할 수 있습니다. 이러한 구현 예제는 다음 두 JavaScript 파일을 참조하십시오.

- connection_ado_conn_string.js 파일은 Configuration/Connections/ASP_Js 폴더에 있습니다.
- connection_common.js 파일은 Configuration/Connections/Shared 폴더에 있습니다.

사용자가 사이트를 열면 Dreamweaver에서 Connections 폴더에 있는 모든 파일을 하나씩 열어 내용을 findConnection()에 전달합니다. 파일의 내용이 유효한 연결 조건과 일치하면 findConnection()에서 연결 객체를 반환합니다. 그러면 Dreamweaver에서는 [데이터베이스] 패널에 모든 연결 객체를 나열합니다.

연결 대화 상자를 열고 새 연결을 만들거나 기존 연결을 복제 또는 편집하도록 선택하면 Dreamweaver에서 inspectConnection() 함수를 호출하여 findConnection()이 만든 것과 동일한 연결 객체를 다시 전달합니다. Dreamweaver에서는 이 과정을 통해 대화 상자를 적절한 연결 정보로 채웁니다.

연결 대화 상자에서 [확인]을 클릭하면 Dreamweaver에서 applyConnection() 함수를 호출하여 HTML을 구성하고 이를 Configuration/Connections 폴더에 있는 연결 포함 파일에 저장합니다. applyConnection() 함수는 필드 중 하나에 오류가 있으므로 대화 상자를 닫을 수 없음을 나타내는 빈 문자열을 반환합니다. 포함 파일은 현재 서버 모델에 대한 기본 파일 확장명 유형을 갖습니다.

사용자가 페이지에 레코드세트나 저장 프로시저와 같이 연결을 사용하는 서버 비헤이비어를 추가하면 Dreamweaver에서 연결 포함 파일을 포함하는 페이지에 명령문을 추가합니다.

findConnection()

지원 버전

Dreamweaver UltraDev 4

설명

Dreamweaver에서는 이 함수를 호출하여 지정된 HTML 소스에서 연결을 검색하고 연결 매개 변수를 파싱합니다. 이 소스 파일의 내용이 유효한 연결 조건과 일치하는 경우 findConnection()은 연결 객체를 반환하고 그렇지 않으면 null 값을 반환합니다.

인수

htmlSource

htmlSource 인수는 연결에 대한 HTML 소스입니다.

반환값

다음 표에 나열된 속성의 특정 조합에 대한 값을 제공하는 연결 객체를 반환합니다. 이 함수가 반환하는 속성 값은 문서 형식에 따라 달라집니다.

속성	설명
name	연결의 이름입니다.
type	useHTTP가 false인 경우 런타임에 데이터베이스에 연결하는 데 사용할 DLL을 나타냅니다.
string	런타임 연결 문자열입니다. ADO의 경우 연결 매개 변수의 문자열이고, JDBC의 경우 연결 URL입니다.
dsn	ODBC 또는 Cold Fusion 런타임 연결에 사용된 데이터 소스 이름입니다.
driver	런타임에 사용된 JDBC 드라이버의 이름입니다.
username	런타임 연결에 사용된 사용자 이름입니다.
password	런타임 연결에 사용된 암호입니다.
designTimeString	디자인 타임 연결 문자열입니다. string을 참조하십시오.
designTimeDsn	디자인 타임 데이터 소스 이름입니다. dsn을 참조하십시오.
designTimeDriver	디자인 타임에 사용된 JDBC 드라이버의 이름입니다.
designTimeUsername	디자인 타임 연결에 사용된 사용자의 이름입니다.
designTimePassword	디자인 타임 연결에 사용된 암호입니다.
designTimeType	디자인 타임 연결 유형입니다.
usesDesignTimeInfo	false이면 Dreamweaver에서 디자인 타임에 런타임 속성을 사용하고 그렇지 않으면 디자인 타임 속성을 사용합니다.

속성	설명
useHTTP	true 또는 false를 포함하는 문자열입니다. true는 디자인 타임에 HTTP 연결을 사용하도록 지정하고 false는 DLL을 사용하도록 지정합니다.
includePattern	[라이브 데이터] 및 [브라우저에서 미리 보기]를 수행할 때 페이지에서 파일 포함 명령문을 찾는 데 사용된 정규식입니다.
variables	각 페이지 변수에 대한 속성이 해당 값으로 설정된 객체입니다. 이 객체는 [라이브 데이터] 및 [브라우저에서 미리 보기]가 수행될 때 사용됩니다.
catalog	표시되는 메타데이터 양을 제한하는 데이터베이스 식별자를 포함하는 문자열입니다.
schema	표시되는 메타데이터 양을 제한하는 데이터베이스 식별자를 포함하는 문자열입니다.
filename	연결을 만드는 데 사용된 대화 상자의 이름입니다.

htmlSource에 연결이 없으면 null 값이 반환됩니다.

참고: 개발자는 *applyConnection()*이 표준 속성과 함께 반환하는 사용자 정의 속성(예: 메타데이터)을 HTML 소스에 추가할 수 있습니다.

inspectConnection()

지원 버전

Dreamweaver UltraDev 4

설명

사용자가 기존 연결을 편집하면 Dreamweaver에서 이 함수를 호출하여 연결을 정의하는 대화 상자 데이터를 초기화합니다. 이 과정을 통해 Dreamweaver에서 대화 상자를 적절한 연결 정보로 채웁니다.

인수

parameters

parameters 인수는 findConnection() 함수에서 반환하는 객체와 동일합니다.

반환값

없음

applyConnection()

지원 버전

Dreamweaver UltraDev 4

설명

사용자가 연결 대화 상자에서 [확인]을 클릭하면 Dreamweaver에서 이 함수를 호출합니다. applyConnection() 함수는 연결에 대한 HTML 소스를 생성합니다. Dreamweaver에서는 HTML을 Configuration/Connections/**connection-name.ext** 포함 파일에 기록합니다. 여기서 **connection-name**은 연결의 이름이고(76페이지의 “[새 연결 유형 개발](#)” 참조) **.ext**는 서버 모델과 연관된 기본 확장명입니다.

인수

없음

반환값

연결의 HTML 소스를 반환합니다. Dreamweaver에서는 연결 대화 상자도 닫습니다. 필드 유효성 검사 오류가 발생하면 applyConnection()은 오류 메시지를 표시하고 대화 상자를 열어 두어야 함을 나타내는 빈 문자열을 반환합니다.

생성된 포함 파일

applyConnection()을 통해 생성되는 포함 파일은 연결의 모든 속성을 선언합니다. 포함 파일의 파일 이름은 연결 이름이며 현재 사이트와 연관된 서버 모델에 정의된 파일 이름 확장명을 갖습니다.

참고: 연결은 공유되므로 *allowMultiple value*를 *false*로 설정하십시오. 이렇게 하면 연결 파일이 문서에 한 번만 포함되고 서버 스크립트가 다른 서버 비헤이비어에서 사용되는 경우 페이지에 그대로 유지됩니다.

다음 단원에서 다양한 기본 서버 모델에 대해 applyConnection()이 생성하는 샘플 포함 파일을 볼 수 있습니다.

참고: 연결 포함 파일 형식을 만들려면 81페이지의 “[연결 유형에 대한 정의 파일](#)”에 설명된 대로 connection_includefile.edml 과 유사한 새로운 EDML 매핑 파일을 정의해야 합니다.

ASP JavaScript

ASP와 JavaScript 포함 파일의 이름은 MyConnection1.asp로 지정해야 합니다. 여기서 MyConnection1은 연결의 이름입니다. 다음 샘플은 ADO 연결 문자열에 대한 포함 파일입니다.

```
<%
    // Filename="Connection_ado_conn_string.htm"
    // Type="ADO"
    // HTTP="true"
    // Catalog=""
    // Schema=""
    var MM_MyConnection1_STRING = "dsn=pubs";
%>
```

서버 비헤이비어 파일은 다음 예제에 표시된 대로 상대적 파일 포함 명령문을 사용하여 이 연결을 포함합니다.

```
<!--#include file="../../../Connections/MyConnection1.asp"-->
```

ColdFusion

UltraDev 4 ColdFusion을 사용하는 경우 Dreamweaver에서 ColdFusion 포함 파일을 사용하여 데이터 소스 목록을 가져옵니다.

참고: 일반 Dreamweaver ColdFusion을 사용하는 경우에는 Dreamweaver에서 모든 포함 파일을 무시하고 대신 RDS를 사용하여 ColdFusion에서 데이터 소스 목록을 가져옵니다.

UltraDev 4 ColdFusion 포함 파일의 이름은 MyConnection1.cfm으로 지정해야 합니다. 여기서 MyConnection1은 연결의 이름입니다. 다음 예제에서는 product 테이블에 대한 ColdFusion 연결의 포함 파일을 보여 줍니다.

```
<!-- FileName="Connection_cf_dsn.htm" dsn=products -->
<!-- Type="ADO" -->
<!-- Catalog="" -->
<!-- Schema="" -->
<!-- HTTP="false" -->
<CFSET MM_MyConnection1_DSN          = "products">
<CFSET MM_MyConnection1_USERNAME    = "">
<CFSET MM_Product_USERNAME          = "">
<CFSET MM_MyConnection1_PASSWORD    = "">
```

서버 비헤이비어 파일은 다음 예제에 표시된 대로 cfinclude 문을 사용하여 이 연결을 포함합니다.

```
<cfinclude template="Connections/MyConnection1.cfm">
```

연결 유형에 대한 정의 파일

각 서버 모델에는 연결 유형을 정의하고 포함 파일에 정의된 속성을 Dreamweaver 인터페이스의 요소에 매핑하는 connection_includefile.edml 파일이 있습니다.

Dreamweaver에서는 다음 표에 나열된 것처럼 미리 정의된 서버 모델마다 하나씩 기본 정의 파일을 제공합니다.

서버 모델	Configuration/Connections 폴더의 하위 폴더
ASP JavaScript	ASP_Js
ASP.NET CSharp	ASP.NET_Csharp
ASP.NET VBScript	ASP.NET_VB
ASP VBScript	ASP_Vbs
ColdFusion	ColdFusion
JavaServer Page	JSP
PHP MySql	PHP_MySql

Dreamweaver에서는 quickSearch 및 searchPattern 매개 변수를 사용하여 연결 블록을 인식하고 insertText 매개 변수를 사용하여 연결 블록을 만듭니다. EDML 태그 및 속성, 정규식 검색 패턴에 대한 자세한 내용은 **Dreamweaver** 확장의 "서버 비헤이비어"를 참조하십시오.

참고: 포함 파일의 형식을 변경하거나 새 서버 모델에 대한 포함 파일을 정의하는 경우에는 Dreamweaver UI, [라이브 데이터] 및 [브라우저에서 미리 보기]를 사용하여 연결 매개 변수를 매핑해야 합니다. 기본 ASP JS 서버 모델과 연관된 다음 샘플 EDML 파일에서는 서버로 페이지를 보내기 전에 개별 실제 값을 가진 모든 연결 페이지 변수를 매핑합니다. EDML 및 정규식 검색 패턴에 대한 자세한 내용은 Dreamweaver 확장의 "서버 비헤이비어"를 참조하십시오.

```
<participant name="connection_includefile" version="5.0">
  <quickSearch>
    <![CDATA[/ HTTP=]]></quickSearch>
    <insertText location="">
<![CDATA[<%
// FileName="@filename@"
// Type="@type@" @@designtimeString@@
// DesigntimeType="@designtimeType@"
// HTTP="@http@"
// Catalog="@catalog@"
// Schema="@schema@"
var MM_@cname@_STRING = @string@@
%>
]]>
    </insertText>
    <searchPatterns whereToSearch="directive">
      <searchPattern paramNames="filename">
        <![CDATA[/\//\s*FileName="(^[^"]*)" /]]></searchPattern>
      <searchPattern paramNames="type,designtimeString">
        <![CDATA[/\//\s+Type="(\\w*)" "([^\r\n]*) /]]></searchPattern>
      <searchPattern paramNames="designtimeType" isOptional="true">
        <![CDATA[/\//\s*DesigntimeType="(\\w*)" /]]></searchPattern>
      <searchPattern paramNames="http">
        <![CDATA[/\//\s*HTTP="(\\w*)" /]]></searchPattern>
      <searchPattern paramNames="catalog">
        <![CDATA[/\//\s*Catalog="(\\w*)" /]]></searchPattern>
      <searchPattern paramNames="schema">
        <![CDATA[/\//\s*Schema="(\\w*)" /]]></searchPattern>
      <searchPattern paramNames="cname,string">
        <![CDATA[/var\s+MM_(\\w*)_STRING\s*=\s*([^\r\n]+)/]]></searchPattern>
    </searchPatterns>
  </participant>
```

이 예제의 @@filename@@과 같은 EDML 파일의 토큰은 포함 파일의 값을 연결 객체의 속성에 매핑합니다. JavaScript 구현 파일에서 연결 객체의 속성을 설정합니다.

Dreamweaver와 함께 제공되는 모든 기본 연결 대화 상자에서는 connection_includefile.edml 매핑 파일을 사용합니다. Dreamweaver에서 이 파일을 찾을 수 있도록 하려면 다음 예제에 표시된 대로 JavaScript 구현 파일에서 해당 이름을 설정합니다.

```
var PARTICIPANT_FILE = "connection_includefile";
```

사용자 정의 연결 유형을 만드는 경우 사용자 정의 대화 상자에서 모든 매핑 파일을 사용할 수 있습니다. 매핑 파일을 만들 경우에는 EDML 파일에 connection_includefile이 아닌 다른 이름을 사용할 수 있습니다. 다른 이름을 사용하는 경우에는 다음 예제에 표시된 대로 PARTICIPANT_FILE 변수에 할당되는 값을 지정할 때 JavaScript 구현 파일에서 이 이름을 사용해야 합니다.

```
var PARTICIPANT_FILE = "myConnection_mappingfile";
```

10장: 소스 제어 통합 API

소스 제어 통합 API를 사용하여 공유 라이브러리를 작성할 수 있습니다. 이러한 API를 사용하면 Sourcesafe 또는 CVS와 같은 소스 제어 시스템을 통해 Adobe® Dreamweaver® 체크 인/체크 아웃 기능을 확장할 수 있습니다.

라이브러리는 Dreamweaver에서 소스 제어 시스템과 통합하기 위한 최소한의 API 함수를 제공해야 합니다. 또한 라이브러리는 Program Files/Adobe/Adobe Dreamweaver CS4/Configuration/SourceControl 폴더에 있어야 합니다.

Dreamweaver는 시작될 때 각 라이브러리를 로드합니다. Dreamweaver에서는 각 API 함수에 대해 GetProcAddress()를 호출하여 라이브러리가 지원하는 기능을 확인합니다. 주소가 없으면 Dreamweaver에서는 라이브러리가 API를 지원하지 않는 것으로 간주합니다. 주소가 있으면 함수의 기능을 지원하기 위해 함수의 라이브러리 버전이 사용됩니다. Dreamweaver 사용자가 사이트를 정의하거나 편집한 다음 [웹 서버 SCS] 탭을 선택하면 DLL에 해당하는 선택 항목이 탭에 표시됩니다. 이러한 선택 항목은 표준 항목과 함께 탭에 나타납니다. DLL은 Program Files/Adobe/Adobe Dreamweaver CS4/Configuration/SourceControl 폴더에서 로드됩니다.

사용자 정의 항목을 추가할 수 있도록 [사이트] > [소스 제어] 메뉴를 만들려면 다음 코드를 추가합니다. menus.xml 파일의 [사이트] 메뉴에서 코드를 추가하십시오.

```
<menu name="Source Control" id="DWMenu_MainSite_Site_Source-
Control"><menuitem dynamic name="None"file="Menus/MM/¬
File_SCSItems.htm" id="DWMenu_MainSite_Site_NewFeatures_¬
Default" />
</menu>
```

Dreamweaver의 소스 제어 통합 작동 방법

Dreamweaver 사용자가 서버 연결, 파일 전송 또는 디자인 노트 기능을 선택하면 Dreamweaver에서 해당 API 함수(Connect(), Disconnect(), Get(), Put(), Checkin(), Checkout(), Undocheckout() 및 Synchronize())의 DLL 버전을 호출합니다. 이 DLL을 통해 정보를 모으거나 사용자가 DLL과 상호 작용할 수 있는 대화 상자를 표시하는 등의 요청이 처리됩니다. 또한 이 DLL에 정보 또는 오류 메시지가 표시됩니다.

소스 제어 시스템은 [디자인 노트]와 [체크 인]/[체크 아웃]을 선택적으로 지원할 수 있습니다. Dreamweaver 사용자는 [사이트 편집] 대화 상자에서 [디자인 노트] 탭을 선택하고 이 기능을 활성화하는 체크 상자를 클릭하여 소스 제어 시스템에서 디자인 노트를 활성화할 수 있습니다. 이 과정은 FTP 및 LAN을 사용하여 디자인 노트를 활성화하는 것과 같습니다. 소스 제어 시스템이 디자인 노트를 지원하지 않는 경우 사용자가 이 기능을 사용하려고 하면 Dreamweaver에서는 FTP와 LAN에서처럼 디자인 노트(MNO) 파일을 전송하여 디자인 노트를 유지 관리합니다.

[체크 인]/[체크 아웃]은 [디자인 노트] 기능과는 다르게 처리됩니다. 소스 제어 시스템이 이 기능을 지원하면 사용자는 [디자인 노트] 대화 상자에서 이 기능을 무시할 수 없습니다. 사용자가 소스 제어 시스템을 무시하려고 하면 오류 메시지가 나타납니다.

소스 제어 시스템 기능 추가

일련의 메뉴 항목과 해당 C 함수를 반환하는 GetNewFeatures 핸들러를 작성하여 Dreamweaver에 소스 제어 시스템 기능을 추가할 수 있습니다. 예를 들어, Sourcesafe 라이브러리를 작성하여 Dreamweaver 사용자가 파일의 작업 내역을 볼 수 있도록 하려면 [작업 내역] 메뉴 항목과 history의 C 함수 이름을 반환하는 GetNewFeatures 핸들러를 작성할 수 있습니다. 이렇게 하면 Windows에서 사용자가 파일을 마우스 오른쪽 버튼으로 클릭할 때 [작업 내역] 메뉴 항목이 메뉴에 표시됩니다. 사용자가 [작업 내역] 메뉴 항목을 선택하면 해당 함수가 호출되고 선택된 파일이 DLL에 전달됩니다. 이 DLL에 의해 [작업 내역] 대화 상자가 표시되어 Sourcesafe와 동일한 방식으로 대화형 기능을 사용할 수 있습니다.

소스 제어 통합 API 필수 함수

소스 제어 통합 API에는 필수 함수와 선택적 함수가 있습니다. 이 단원에서는 필수 함수에 대해 살펴봅니다.

bool SCS_GetAgentInfo()

설명

이 함수는 DLL이 해당 이름과 설명을 반환하도록 요청하며 이러한 이름과 설명은 [사이트 편집] 대화 상자에 나타납니다. 이름 (예: Sourcesafe, WebDav, Perforce)은 [서버 액세스] 팝업 메뉴에 나타나고 설명은 팝업 메뉴 아래에 나타납니다.

인수

char name[32], char version[32], char description[256], const char *dwAppVersion

- **name** 인수는 소스 제어 시스템의 이름입니다. 이 이름은 [사이트 편집] 대화 상자의 [소스 제어] 탭에서 소스 제어 시스템을 선택하는 콤보 상자에 나타납니다. 이름에는 최대 32자를 사용할 수 있습니다.
- **version** 인수는 DLL의 버전을 나타내는 문자열입니다. 버전은 [사이트 편집] 대화 상자의 [소스 제어] 탭에 나타납니다. 버전에는 최대 32자를 사용할 수 있습니다.
- **description** 인수는 소스 제어 시스템에 대한 설명을 나타내는 문자열입니다. 설명은 [사이트 편집] 대화 상자의 [소스 제어] 탭에 나타납니다. 설명에는 최대 256자를 사용할 수 있습니다.
- **dwAppVersion** 인수는 DLL을 호출하는 Dreamweaver의 버전을 나타내는 문자열입니다. DLL은 이 문자열을 사용하여 Dreamweaver의 버전과 언어를 확인할 수 있습니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_Connect()

설명

이 함수는 사용자를 소스 제어 시스템에 연결합니다. DLL에 로그인 정보가 없으면 DLL은 대화 상자를 표시하여 사용자에게 정보 입력을 요구하고 다음에 사용할 수 있도록 데이터를 저장해야 합니다.

인수

void **connectionData, const char siteName[64]

- **connectionData** 인수는 다른 API 함수를 호출할 때 에이전트가 Dreamweaver에게 전달하도록 요청하는 데이터의 핸들입니다.
- **siteName** 인수는 사이트의 이름을 가리키는 문자열입니다. 사이트 이름은 최대 64자일 수 있습니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_Disconnect()

설명

이 함수는 소스 제어 시스템에 연결된 사용자의 연결을 해제합니다.

인수**void *connectionData**

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_IsConnected()**설명**

이 함수는 연결 상태를 확인합니다.

인수**void *connectionData**

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

int SCS_GetRootFolderLength()**설명**

이 함수는 루트 폴더 이름의 길이를 반환합니다.

인수**void *connectionData**

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.

반환값

루트 폴더 이름의 길이를 나타내는 정수를 반환합니다. 함수가 <0을 반환하면 오류로 간주되며 DLL에서 오류 메시지를 가져올 수 있으면 가져옵니다.

bool SCS_GetRootFolder()**설명**

이 함수는 루트 폴더의 이름을 반환합니다.

인수**void *connectionData, char remotePath[], const int folderLen**

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePath**는 루트 폴더의 전체 원격 경로가 저장되는 버퍼입니다.
- **folderLen** 인수는 **remotePath**의 길이를 나타내는 정수입니다. 이 값은 **GetRootFolderLength**가 반환합니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

int SCS_GetFolderListLength()

설명

이 함수는 전달된 폴더의 항목 수를 반환합니다.

인수

void *connectionData, const char *remotePath

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePath** 인수는 DLL이 항목의 수를 확인하는 원격 폴더의 전체 경로 및 이름입니다.

반환값

현재 폴더에 있는 항목의 수를 나타내는 정수를 반환합니다. 함수가 <0을 반환하면 오류로 간주되며 DLL에서 오류 메시지를 가져올 수 있으면 가져옵니다.

bool SCS_GetFolderList()

설명

이 함수는 전달된 폴더 내의 파일 및 폴더의 목록을 반환합니다. 여기에는 수정 날짜, 크기, 해당 항목이 폴더인지, 아니면 파일인지 등과 같은 관련 정보가 포함됩니다.

인수

void *connectionData, const char *remotePath, itemInfo itemList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePath** 인수는 DLL이 항목의 수를 확인하는 원격 폴더의 경로입니다.
- **itemList** 인수는 itemInfo 구조체의 미리 할당된 목록입니다.

이름	char[256]	파일 또는 폴더의 이름
isFolder	bool	폴더이면 true, 파일이면 false
month	int	수정 날짜의 월 구성 요소(1-12)
day	int	수정 날짜의 일 구성 요소(1-31)
year	int	수정 날짜의 연도 구성 요소(1900+)
hour	int	수정 날짜의 시 구성 요소(0-23)
minutes	int	수정 날짜의 분 구성 요소(0-59)
seconds	int	수정 날짜의 초 구성 요소(0-59)
type	char[256]	파일의 형식(DLL에서 설정하지 않은 경우 Dreamweaver에서는 파일 확장명을 사용하여 형식 확인)
size	int	바이트 단위

- **numItems** 인수는 itemList에 할당된 항목의 수입니다(GetFolderListLength에서 반환됨).

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_Get()**설명**

이 함수는 파일이나 폴더의 목록을 가져와 로컬에 저장합니다.

인수

void *connectionData, const char *remotePathList[], const char *localPathList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePathList** 인수는 전체 경로 및 이름으로 지정된 가져올 원격 파일 또는 폴더의 목록입니다.
- **localPathList** 인수는 로컬 파일 이름 또는 폴더 경로의 미리링된 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_Put()**설명**

이 함수는 로컬 파일이나 폴더의 목록을 소스 제어 시스템에 올립니다.

인수

void *connectionData, const char *localPathList[], const char *remotePathList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **localPathList** 인수는 소스 제어 시스템에 올릴 로컬 파일 이름 또는 폴더 경로의 목록입니다.
- **remotePathList** 인수는 원격 파일 이름 또는 폴더 경로의 미리링된 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_NewFolder()**설명**

이 함수는 새 폴더를 만듭니다.

인수

void *connectionData, const char *remotePath

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePath** 인수는 DLL에 의해 만들어지는 원격 폴더의 전체 경로입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_Delete()

설명

이 함수는 소스 제어 시스템에서 파일이나 폴더의 목록을 삭제합니다.

인수

void *connectionData, const char *remotePathList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePathList** 인수는 삭제할 원격 파일 이름 또는 폴더 경로의 목록입니다.
- **numItems** 인수는 **remotePathList**에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_Rename()

설명

이 함수는 **oldRemotePath** 및 **newRemotePath**에 지정된 값에 따라 파일 또는 폴더의 이름을 변경하거나 파일 또는 폴더를 이동합니다. 예를 들어, **oldRemotePath**가 "\$/folder1/file1"과 같고 **newRemotePath**가 "\$/folder1/renamefile1"과 같으면, file1은 renamefile1로 이름이 변경되고 folder1로 이동합니다.

oldRemotePath가 "\$/folder1/file1"과 같고 **newRemotePath**가 "\$/folder1/subfolder1/file1"과 같으면, file1이 subfolder1 폴더로 이동합니다.

이 함수의 호출이 이동을 위한 것인지, 아니면 이름 변경을 위한 것인지를 알아내려면 두 입력 값의 부모 경로를 확인합니다. 두 경로가 동일하면 이름 변경 작업이 수행됩니다.

인수

void *connectionData, const char *oldRemotePath, const char *newRemotePath

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **oldRemotePath** 인수는 이름을 변경할 원격 파일 또는 폴더 경로입니다.
- **newRemotePath** 인수는 파일 또는 폴더에 대한 새로운 이름의 원격 경로입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_ItemExists()

설명

이 함수는 서버에 파일이나 폴더가 있는지 여부를 확인합니다.

인수

void *connectionData, const char *remotePath

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePath** 인수는 원격 파일 또는 폴더 경로입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

소스 제어 통합 API 선택적 함수

소스 제어 통합 API에는 필수 함수와 선택적 함수가 있습니다. 이 단원에서는 선택적인 함수에 대해 살펴봅니다.

bool SCS_GetConnectionInfo()

설명

이 함수는 대화 상자를 표시하여 사용자가 이 사이트의 연결 정보를 변경하거나 설정할 수 있도록 하지만 연결하지는 않습니다. 이 함수는 사용자가 [사이트 편집] 대화 상자의 [원격 정보] 섹션에 있는 [설정] 버튼을 클릭할 때 호출됩니다.

인수

void **connectionData, const char siteName[64]

- **connectionData** 인수는 다른 API 함수 호출 시 에이전트가 Dreamweaver에게 전달하도록 요청하는 데이터 핸들입니다.
- **siteName** 인수는 사이트의 이름을 가리키는 문자열입니다. 이름은 64자를 초과할 수 없습니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_SiteDeleted()

설명

이 함수는 사이트가 삭제되었거나 소스 제어 시스템에 더 이상 결합되어 있지 않다는 것을 DLL에 알립니다. 이는 소스 제어 시스템이 사이트의 영구적인 정보를 삭제할 수 있음을 나타냅니다.

인수

const char siteName[64]

- **siteName** 인수는 사이트의 이름을 가리키는 문자열입니다. 이름은 64자를 초과할 수 없습니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_SiteRenamed()

설명

이 함수는 사이트에 대한 영구적인 정보를 업데이트할 수 있도록 사용자가 사이트 이름을 변경했음을 DLL에 알립니다.

인수

const char oldSiteName[64], const char newSiteName[64]

- **oldSiteName** 인수는 이름이 변경되기 전의 사이트의 원래 이름을 가리키는 문자열입니다. 이름은 64자를 초과할 수 없습니다.
- **newSiteName** 인수는 이름이 변경된 후의 사이트의 새 이름을 가리키는 문자열입니다. 이름은 64자를 초과할 수 없습니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

int SCS_GetNumNewFeatures()

설명

이 함수는 Dreamweaver에 추가되는 새 기능(예: 파일 작업 내역, 차이 등)의 수를 반환합니다.

인수

없음

반환값

Dreamweaver에 추가되는 새 기능의 수를 나타내는 정수를 반환합니다. 함수가 <0을 반환하면 오류로 간주되며 DLL에서 오류 메시지를 가져올 수 있으면 가져옵니다.

bool SCS_GetNewFeatures()

설명

이 함수는 Dreamweaver의 기본 메뉴와 컨텍스트 메뉴에 추가할 메뉴 항목의 목록을 반환합니다. 예를 들어, Sourcesafe DLL은 기본 메뉴에 [작업 내역]과 [파일 비교]를 추가할 수 있습니다.

인수

char menuItemList[][32], scFunction functionList[], scFunction enablerList[], const int numNewFeatures

- **menuItemList** 인수는 DLL에 의해 채워지는 문자열 목록이며, 기본 메뉴와 컨텍스트 메뉴에 추가할 메뉴 항목을 지정합니다. 각 문자열은 최대 32자를 포함할 수 있습니다.
- **functionList** 인수는 DLL에 의해 채워지며, 사용자가 해당 메뉴 항목을 선택할 때 호출할 DLL의 루틴을 지정합니다.
- **enablerList** 인수는 DLL에 의해 채워지며, Dreamweaver에서 해당 메뉴 항목의 활성화 여부를 확인해야 할 때 호출할 DLL의 루틴을 지정합니다.
- **numNewFeatures** 인수는 DLL에 의해 추가될 항목의 수입니다. 이 값은 GetNumNewFeatures() 호출에서 가져옵니다.

다음 함수 서명은 functionlist 및 enablerList 인수로 SCS_GetNewFeatures() 호출에 전달되는 함수 및 활성화자를 정의합니다.

```
bool (*scFunction)(void *connectionData, const char *remotePathList[],
    const char *localPathList[], const int numItems)
```

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_GetCheckoutName()

설명

이 함수는 현재 사용자의 체크 아웃 이름을 반환합니다. 소스 제어 시스템이 이 기능을 지원하지 않는 경우 사용자가 해당 기능을 활성화하면 이 함수는 Dreamweaver의 내부 체크 인/체크 아웃 기능을 사용하여 LCK 파일을 소스 제어 시스템과 주고 받게 됩니다.

인수

void *connectionData, char checkOutName[64], char emailAddress[64]

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **checkOutName** 인수는 현재 사용자의 이름입니다.
- **emailAddress** 인수는 현재 사용자의 전자 메일 주소입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_Checkin()

설명

이 함수는 로컬 파일이나 폴더의 목록을 소스 제어 시스템으로 체크 인합니다. DLL에 의해 파일이 읽기 전용으로 만들어집니다. 소스 제어 시스템이 이 기능을 지원하지 않는 경우 사용자가 해당 기능을 활성화하면 이 함수는 Dreamweaver의 내부 체크 인/체크 아웃 기능을 사용하여 LCK 파일을 소스 제어 시스템과 주고 받게 됩니다.

인수

void *connectionData, const char *localPathList[], const char *remotePathList[], bool successList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **localPathList** 인수는 체크 인할 로컬 파일 이름 또는 폴더 경로의 목록입니다.
- **remotePathList** 인수는 원격 파일 이름 또는 폴더 경로의 미러링된 목록입니다.
- **successList** 인수는 Dreamweaver에서 성공적으로 체크 인된 파일을 인식할 수 있도록 DLL에 의해 채워지는 부울 값의 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_Checkout()

설명

이 함수는 소스 제어 시스템에서 로컬 파일이나 폴더의 목록을 체크 아웃합니다. 이 DLL에 의해 파일에 쓸 수 있는 권한이 부여됩니다. 소스 제어 시스템이 이 기능을 지원하지 않는 경우 사용자가 해당 기능을 활성화하면 이 함수는 Dreamweaver의 내부 체크 인/체크 아웃 기능을 사용하여 LCK 파일을 소스 제어 시스템과 주고 받게 됩니다.

인수

void *connectionData, const char *remotePathList[], const char *localPathList[], bool successList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePathList** 인수는 체크 아웃할 원격 파일 이름 또는 폴더 경로의 목록입니다.
- **localPathList** 인수는 로컬 파일 이름 또는 폴더 경로의 미리링된 목록입니다.
- **successList** 인수는 Dreamweaver에서 성공적으로 체크 아웃된 파일을 인식할 수 있도록 DLL에 의해 채워지는 부울 값의 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_UndoCheckout()

설명

이 함수는 파일 또는 폴더의 목록에 대한 체크 아웃 상태를 실행 취소합니다. DLL에 의해 파일이 읽기 전용으로 만들어집니다. 소스 제어 시스템이 이 기능을 지원하지 않는 경우 사용자가 해당 기능을 활성화하면 이 함수는 Dreamweaver의 내부 체크 인/체크 아웃 기능을 사용하여 LCK 파일을 소스 제어 시스템과 주고 받게 됩니다.

인수

void *connectionData, const char *remotePathList[], const char *localPathList[], bool successList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePathList** 인수는 체크 아웃을 취소할 원격 파일 이름이나 폴더 경로의 목록입니다.
- **localPathList** 인수는 로컬 파일 이름 또는 폴더 경로의 미리링된 목록입니다.
- **successList** 인수는 Dreamweaver에서 체크 아웃이 성공적으로 취소된 파일을 인식할 수 있도록 DLL에 의해 채워지는 부울 값의 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

int SCS_GetNumCheckedOut()

설명

이 함수는 파일을 체크 아웃한 사용자의 수를 반환합니다.

인수

void *connectionData, const char *remotePath

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePath** 인수는 체크 아웃한 사용자의 수를 확인할 대상 원격 파일 또는 폴더 경로입니다.

반환값

파일을 체크 아웃한 사용자의 수를 나타내는 정수를 반환합니다. 함수가 < 0을 반환하면 오류로 간주되며 DLL에서 오류 메시지를 가져올 수 있으면 가져옵니다.

bool SCS_GetFileCheckoutList()

설명

이 함수는 파일을 체크 아웃한 사용자의 목록을 반환합니다. 목록이 비어 있으면 파일을 체크 아웃한 사용자가 없는 것입니다.

인수

void *connectionData, const char *remotePath, char checkOutList[][64], char emailAddressList[][64], const int numCheckedOut

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePath** 인수는 체크 아웃한 사용자의 수를 확인할 대상 원격 파일 또는 폴더 경로입니다.
- **checkOutList** 인수는 파일을 체크 아웃한 사용자에게 해당하는 문자열의 목록입니다. 각 사용자 문자열의 최대 길이는 64자를 초과할 수 없습니다.
- **emailAddressList** 인수는 사용자의 전자 메일 주소에 해당하는 문자열의 목록입니다. 각 전자 메일 주소 문자열의 최대 길이는 64자를 초과할 수 없습니다.
- **numCheckedOut** 인수는 파일을 체크 아웃한 사용자의 수입니다. 이 값은 GetNumCheckedOut()에서 반환됩니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

int SCS_GetErrorMessageLength()

설명

이 함수는 DLL의 현재 내부 오류 메시지 길이를 반환합니다. 이 길이에 따라 GetErrorMessage() 함수에 전달되는 버퍼가 할당됩니다. 이 함수는 API 함수가 실패를 나타내는 false 또는 < 0을 반환하는 경우에만 호출해야 합니다.

인수

void *connectionData

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.

반환값

오류 메시지의 길이를 나타내는 정수를 반환합니다.

bool SCS_GetErrorMessage()

설명

이 함수는 마지막 오류 메시지를 반환합니다. `getErrorMessage()`를 구현하는 경우 Dreamweaver에서는 해당 API 함수가 `false`를 반환할 때마다 이 함수를 호출합니다.

루틴에서 -1 또는 `false`를 반환하면 오류 메시지를 사용할 수 있어야 합니다.

인수

`void *connectionData, char errorMsg[], const int *msgLength`

- `connectionData` 인수는 `Connect()`를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- `errorMsg` 인수는 오류 메시지에 채울 DLL의 미리 할당된 문자열입니다.
- `msgLength` 인수는 `errorMsg[]` 인수로 표시되는 버퍼의 길이입니다.

반환값

부울 값, 성공적으로 실행되면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

int SCS_GetNoteCount()

설명

이 함수는 지정된 원격 파일이나 폴더 경로에 대한 디자인 노트 키의 수를 반환합니다. 소스 제어 시스템에서 지원되지 않는 경우 Dreamweaver에서는 이 정보를 상대 MNO 파일에서 가져옵니다.

인수

`void *connectionData, const char *remotePath`

- `connectionData` 인수는 `Connect()`를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- `remotePath` 인수는 DLL이 첨부된 디자인 노트 수를 확인하는 대상 원격 파일 또는 폴더 경로입니다.

반환값

이 파일과 연관된 디자인 노트의 수를 나타내는 정수를 반환합니다. 함수가 < 0을 반환하면 오류로 간주되며 DLL에서 오류 메시지를 가져올 수 있으면 가져옵니다.

int SCS_GetMaxNoteLength()

설명

이 함수는 지정된 파일이나 폴더에 대한 최대 디자인 노트의 길이를 반환합니다. 소스 제어 시스템에서 지원되지 않는 경우 Dreamweaver는 이 정보를 상대 MNO 파일에서 가져옵니다.

인수

`void *connectionData, const char *remotePath`

- `connectionData` 인수는 `Connect()`를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.

- **remotePath** 인수는 DLL이 최대 디자인 노트 길이를 확인하는 대상 원격 파일 또는 폴더 경로입니다.

반환값

현재 파일과 연관된 가장 긴 디자인 노트의 길이를 나타내는 정수를 반환합니다. 함수가 < 0을 반환하면 오류로 간주되며 DLL에서 오류 메시지를 가져올 수 있으면 가져옵니다.

bool SCS_GetDesignNotes()

설명

이 함수는 지정된 파일이나 폴더의 메타 정보에서 키/값 쌍을 가져옵니다. 소스 제어 시스템에서 지원되지 않는 경우 Dreamweaver는 이 정보를 상대 MNO 파일에서 가져옵니다.

인수

void *connectionData, const char *remotePath, char keyList[][64], char *valueList[], bool showColumnList[], const int noteCount, const int noteLength

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePath** 인수는 DLL이 항목의 수를 확인하는 대상 원격 파일 또는 폴더 경로입니다.
- **keyList** 인수는 "Status"와 같은 디자인 노트 키의 목록입니다.
- **valueList** 인수는 "Awaiting Signoff"와 같은 디자인 노트 키에 해당하는 디자인 노트 값의 목록입니다.
- **showColumnList** 인수는 디자인 노트 키에 해당하는 부울 값의 목록이며, Dreamweaver에서 키를 [사이트] 패널의 열로 표시할 수 있는지 여부를 나타냅니다.
- **noteCount** 인수는 파일 또는 폴더에 첨부된 디자인 노트의 수이며, 이 값은 GetNoteCount() 호출에서 반환됩니다.
- **noteLength** 인수는 디자인 노트의 최대 길이이며, 이 값은 GetMaxNoteLength() 호출에서 반환됩니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_SetDesignNotes()

설명

이 함수는 키/값 쌍을 지정된 파일이나 폴더의 메타 정보에 저장합니다. 이는 파일의 메타 정보 설정을 바꿉니다. 소스 제어 시스템에서 지원되지 않는 경우 Dreamweaver에서는 디자인 노트를 MNO 파일에 저장합니다.

인수

void *connectionData, const char *remotePath, const char keyList[][64], const char *valueList[], bool showColumnList[], const int noteCount, const int noteLength

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePath** 인수는 DLL이 항목의 수를 확인하는 대상 원격 파일 또는 폴더 경로입니다.
- **keyList** 인수는 "Status"와 같은 디자인 노트 키의 목록입니다.
- **valueList** 인수는 "Awaiting Signoff"와 같은 디자인 노트 키에 해당하는 디자인 노트 값의 목록입니다.
- **showColumnList** 인수는 디자인 노트 키에 해당하는 부울 값의 목록이며, Dreamweaver에서 키를 [사이트] 패널의 열로 표시할 수 있는지 여부를 나타냅니다.

- **noteCount** 인수는 파일 또는 폴더에 첨부된 디자인 노트의 수입입니다. DLL은 이 값을 통해 지정된 목록의 크기를 인식합니다. **noteCount**의 값이 0이면 파일에서 모든 디자인 노트가 제거됩니다.
- **noteLength** 인수는 지정된 파일이나 폴더에 대한 가장 큰 디자인 노트의 길이입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_IsRemoteNewer()**설명**

이 함수는 원격 사본이 더 최신 버전인지 확인하기 위해 지정된 각 원격 경로를 확인합니다. 소스 제어 시스템에서 지원되지 않는 경우 Dreamweaver에서는 내부 isRemoteNewer 알고리즘을 사용합니다.

인수

void *connectionData, const char *remotePathList[], const char *localPathList[], int remoteIsNewerList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePathList** 인수는 최신 상태를 비교할 원격 파일 이름 또는 폴더 경로의 목록입니다.
- **localPathList** 인수는 로컬 파일 이름 또는 폴더 경로의 미리링된 목록입니다.
- **remoteIsNewerList** 인수는 Dreamweaver에서 원격측에 있는 해당 파일 중 더 최신인 파일을 인식할 수 있도록 DLL에 의해 채워지는 정수의 목록입니다. 유효한 값은 1, -1 또는 0이며, 1이면 원격 버전이 더 최신임을 나타내고, -1이면 로컬 버전이 더 최신임을 나타냅니다. 0이면 두 버전은 같은 것입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

활성자

선택적 활성자가 소스 제어 시스템에서 지원되지 않거나 응용 프로그램이 서버에 연결되어 있지 않은 경우 Dreamweaver에서는 원격 파일에 대한 정보에 따라 메뉴 항목이 활성화되는 시기를 확인합니다.

bool SCS_canConnect()**설명**

이 함수는 [연결] 메뉴 항목을 활성화할지 여부를 반환합니다.

인수

없음

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_canGet()

설명

이 함수는 [가져오기] 메뉴 항목을 활성화할지 여부를 반환합니다.

인수

void *connectionData, const char *remotePathList[], const char *localPathList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePathList** 인수는 가져올 원격 파일 이름 또는 폴더 경로의 목록입니다.
- **localPathList** 인수는 로컬 파일 이름 또는 폴더 경로의 미리링된 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_canCheckout()

설명

이 함수는 [체크 아웃] 메뉴 항목을 활성화할지 여부를 반환합니다.

인수

void *connectionData, const char *remotePathList[], const char *localPathList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePathList** 인수는 체크 아웃할 원격 파일 이름 또는 폴더 경로의 목록입니다.
- **localPathList** 인수는 로컬 파일 이름 또는 폴더 경로의 미리링된 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_canPut()

설명

이 함수는 [올리기] 메뉴 항목을 활성화할지 여부를 반환합니다.

인수

void *connectionData, const char *localPathList[], const char *remotePathList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **localPathList** 인수는 소스 제어 시스템에 올릴 로컬 파일 이름 또는 폴더 경로의 목록입니다.
- **remotePathList** 인수는 소스 제어 시스템에 올릴 원격 파일 이름 또는 폴더 경로의 미리링된 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_canCheckin()**설명**

이 함수는 [체크 인] 메뉴 항목을 활성화할지 여부를 반환합니다.

인수

void *connectionData, const char *localPathList[], const char *remotePathList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **localPathList** 인수는 체크 인할 로컬 파일 이름 또는 폴더 경로의 목록입니다.
- **remotePathList** 인수는 원격 파일 이름 또는 폴더 경로의 미리링된 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_CanUndoCheckout()**설명**

이 함수는 [체크 아웃 취소] 메뉴 항목을 활성화할지 여부를 반환합니다.

인수

void *connectionData, const char *remotePathList[], const char *localPathList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePathList** 인수는 체크 아웃할 원격 파일 이름 또는 폴더 경로의 목록입니다.
- **localPathList** 인수는 소스 제어 시스템에 올릴 로컬 파일 이름 또는 폴더 경로의 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_canNewFolder()**설명**

이 함수는 [새 폴더] 메뉴 항목을 활성화할지 여부를 반환합니다.

인수

void *connectionData, const char *remotePath

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.

- **remotePath** 인수는 새 폴더가 만들어질 위치를 나타내기 위해 사용자가 선택한 원격 파일 이름 또는 폴더 경로의 목록입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_canDelete()

설명

이 함수는 [삭제] 메뉴 항목을 활성화할지 여부를 반환합니다.

인수

void *connectionData, const char *remotePathList[], const int numItems

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePathList** 인수는 삭제할 원격 파일 이름 또는 폴더 경로의 목록입니다.
- **numItems** 인수는 각 목록에 들어 있는 항목의 수입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_canRename()

설명

이 함수는 [이름 변경] 메뉴 항목을 활성화할지 여부를 반환합니다.

인수

void *connectionData, const char *remotePath

- **connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.
- **remotePathList** 인수는 이름을 변경할 수 있는 원격 파일 이름 또는 폴더 경로입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

bool SCS_BeforeGet()

설명

Dreamweaver에서는 하나 이상의 파일을 가져오거나 체크 아웃하기 전에 이 함수를 호출합니다. 이 함수를 통해 DLL은 여러 파일에 체크 아웃 주석 추가 등과 같은 단일 작업을 수행합니다.

인수

***connectionData**

- ***connectionData** 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

Dreamweaver에서는 다음 순서로 DLL을 호출하여 여러 파일을 가져옵니다.

```
SCS_BeforeGet (connectionData) ;  
SCS_Get (connectionData,remotePathList1,localPathList1,successList1) ;  
SCS_Get (connectionData,remotePathList2,localPathList2,successList2) ;  
SCS_Get (connectionData,remotePathList3,localPathList3,successList3) ;  
SCS_AfterGet (connectionData) ;
```

bool SCS_BeforePut()

설명

Dreamweaver에서는 하나 이상의 파일을 올리거나 체크 인하기 전에 이 함수를 호출합니다. 이 함수를 통해 DLL은 여러 파일에 체크 인 주석 추가 등과 같은 단일 작업을 수행합니다.

인수

*connectionData

- *connectionData 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

Dreamweaver에서는 다음 순서로 DLL을 호출하여 여러 파일을 가져옵니다.

```
SCS_BeforePut (connectionData) ;  
SCS_Put (connectionData,localPathList1,remotePathList1,successList1) ;  
SCS_Put (connectionData,localPathList2,remotePathList2,successList2) ;  
SCS_Put (connectionData,localPathList3,remotePathList3,successList3) ;  
SCS_AfterPut (connectionData) ;
```

bool SCS_AfterGet()

설명

Dreamweaver에서는 하나 이상의 파일을 가져오거나 체크 아웃한 후 이 함수를 호출합니다. DLL은 이 함수를 통해 일괄 가져오기 또는 체크 아웃 이후 요약 대화 상자 만들기 등의 작업을 수행할 수 있습니다.

인수

*connectionData

- *connectionData 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

99페이지의 “bool SCS_BeforeGet()”을 참조하십시오.

bool SCS_AfterPut()

설명

Dreamweaver에서는 하나 이상의 파일을 올리거나 체크 인한 후 이 함수를 호출합니다. DLL은 이 함수를 통해 일괄 올리기 또는 체크 인 이후 요약 대화 상자 만들기 등의 작업을 수행할 수 있습니다.

인수

*connectionData

- *connectionData 인수는 Connect()를 호출하는 동안 Dreamweaver로 전달되는 에이전트 데이터에 대한 포인터입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

100페이지의 “[bool SCS_BeforePut\(\)](#)”을 참조하십시오.

11장: 응용 프로그램

응용 프로그램 함수는 Adobe® Dreamweaver®와 다른 응용 프로그램의 상호 작용과 관련된 작업이나 개별 문서와 독립적인 Dreamweaver 작업을 수행합니다. 예를 들어, 환경 설정, 기존 Dreamweaver 등을 설정하는 작업을 수행합니다.

외부 응용 프로그램 함수

외부 응용 프로그램 함수는 Adobe® Flash® 등의 응용 프로그램과 관련된 작업 및 [브라우저에서 미리 보기]와 [외부 편집기] 환경 설정에 정의된 브라우저 및 외부 편집기와 관련된 작업을 처리합니다. 이 함수를 사용하면 이러한 외부 응용 프로그램에 대한 정보를 가져오고 해당 프로그램을 사용하여 파일을 열 수 있습니다.

dreamweaver.browseDocument()

지원 버전

Dreamweaver 2, Dreamweaver 3 및 4에서 기능이 향상됨

설명

지정된 브라우저에서 지정된 URL을 엽니다.

인수

fileName, {browser}

- fileName 인수는 열 파일의 이름이며, 절대 URL로 표시됩니다.
- browser 인수는 브라우저를 지정합니다. 이 인수는 [브라우저에서 미리 보기] 환경 설정에 정의된 브라우저 이름이거나 primary 또는 secondary가 될 수 있습니다. 이 인수를 생략하면 해당 URL은 사용자의 기본 브라우저에서 열립니다.

참고: "Configuration/ExtensionHelp/browseHelp.htm#helpyou"와 같이 URL에 앵커가 포함되면 일부 브라우저에서 파일을 찾을 수 없습니다.

반환값

없음

예제

다음 함수는 dreamweaver.browseDocument() 함수를 사용하여 브라우저에서 Adobe 홈 페이지를 엽니다.

```
function goToadobe() {
    dreamweaver.browseDocument('http://www.adobe.com/');
}
```

Dreamweaver 4에서는 이 작업을 확장하여 다음 코드로 Microsoft Internet Explorer에서 문서를 열 수 있습니다.

```
function goToadobe() {
    var prevBrowsers = dw.getBrowserList();
    var theBrowser = "";
    for (var i=1; i < prevBrowsers.length; i+2) {
        if (prevBrowsers[i].indexOf('Iexplore.exe') != -1) {
            theBrowser = prevBrowsers[i];
            break;
        }
    }
    dw.browseDocument('http://www.adobe.com/', theBrowser);
}
```

dreamweaver.getBrowserList() 함수에 대한 자세한 내용은 103페이지의 “[dreamweaver.getBrowserList\(\)](#)”를 참조하십시오.

dreamweaver.getBrowserList()

지원 버전

Dreamweaver 3

설명

[파일] > [브라우저에서 미리 보기] 하위 메뉴의 모든 브라우저 목록을 가져옵니다.

인수

없음

반환값

목록의 각 브라우저에 대한 한 쌍의 문자열을 포함하는 배열을 반환합니다. 각 쌍의 첫 번째 문자열은 브라우저의 이름이고, 두 번째 문자열은 사용자 컴퓨터에서의 위치로서 file:// URL로 표시됩니다. 브라우저가 하위 메뉴에 나타나지 않으면 반환되는 값이 없습니다.

dreamweaver.getExtensionEditorList()

지원 버전

Dreamweaver 3

설명

[외부 편집기] 환경 설정에서 지정된 파일의 편집기 목록을 가져옵니다.

인수

fileURL

- fileURL 인수는 전체 file:// URL, 파일 이름 또는 파일 확장명(마침표 포함)이 될 수 있습니다.

반환값

목록의 각 편집기에 대한 한 쌍의 문자열을 포함하는 배열을 반환합니다. 각 쌍의 첫 번째 문자열은 편집기의 이름이고, 두 번째 문자열은 사용자 컴퓨터에서의 위치로서 file:// URL로 표시됩니다. [환경 설정]에 편집기가 나타나지 않으면 함수는 빈 문자열 하나로 구성된 배열을 반환합니다.

예제

`dreamweaver.getExtensionEditorList(".gif")` 함수를 호출하면 다음 문자열을 포함하는 배열이 반환될 수 있습니다.

- "Fireworks 3"
- "file:///C:/Program Files/Adobe/Fireworks 3/Fireworks 3.exe"

dreamweaver.getExternalTextEditor()**지원 버전**

Dreamweaver 4

설명

현재 구성된 외부 텍스트 편집기의 이름을 가져옵니다.

인수

없음

반환값

전체 경로가 아니라 UI(사용자 인터페이스)로 표시하기에 적합한 텍스트 편집기 이름을 포함하는 문자열을 반환합니다.

dreamweaver.getFlashPath()**지원 버전**

Dreamweaver MX

설명

Flash MX 응용 프로그램의 전체 경로를 파일 URL 형식으로 가져옵니다.

인수

없음

반환값

두 요소를 포함하는 배열을 반환합니다. 요소 [0]은 Flash MX 편집기의 이름을 포함하는 문자열입니다. 요소 [1]은 로컬 컴퓨터에서의 Flash 응용 프로그램 경로를 포함하는 문자열이며, file:// URL로 표시됩니다. Flash가 설치되어 있지 않으면 반환되는 값이 없습니다.

예제

다음 예제에서는 `dw.getFlashPath()` 함수를 호출하여 Flash 응용 프로그램의 경로를 얻은 다음 경로를 file://URL 형식으로 `dw.openWithApp()` 함수에 전달하여 Flash에서 문서를 엽니다.

```
var myDoc = dreamweaver.getDocumentDOM();

if (dreamweaver.validateFlash()) {
    var flashArray = dreamweaver.getFlashPath();
    dreamweaver.openWithApp(myDoc.myForm.swfFilePath, flashArray[1]);
}
```

dreamweaver.getPrimaryBrowser()

지원 버전

Dreamweaver 3

설명

기본 브라우저의 경로를 가져옵니다.

인수

없음

반환값

사용자 컴퓨터에서의 기본 브라우저 경로가 포함된 문자열을 반환합니다. 이 경로는 file:// URL로 표시됩니다. 기본 브라우저가 정의되어 있지 않으면 반환되는 값이 없습니다.

dreamweaver.getPrimaryExtensionEditor()

지원 버전

Dreamweaver 3

설명

지정된 파일의 기본 편집기를 가져옵니다.

인수

fileURL

- fileURL 인수는 열 파일의 경로이며, file:// URL로 표시됩니다.

반환값

한 쌍의 문자열을 포함하는 배열을 반환합니다. 각 쌍의 첫 번째 문자열은 편집기의 이름이고, 두 번째 문자열은 사용자 컴퓨터에서의 위치로서 file:// URL로 표시됩니다. 기본 편집기가 정의되어 있지 않으면 함수는 빈 문자열 하나로 구성된 배열을 반환합니다.

dreamweaver.getSecondaryBrowser()

지원 버전

Dreamweaver 3

설명

보조 브라우저의 경로를 가져옵니다.

인수

없음

반환값

사용자 컴퓨터에서의 보조 브라우저 경로가 포함된 문자열을 반환합니다. 이 경로는 `file://` URL로 표시됩니다. 보조 브라우저가 정의되어 있지 않으면 반환되는 값이 없습니다.

dreamweaver.openHelpURL()**지원 버전**

Dreamweaver MX

설명

지정된 도움말 파일을 운영 체제의 도움말 뷰어에서 엽니다.

Dreamweaver는 브라우저가 아닌 표준 운영 체제의 도움말 뷰어에 도움말 내용을 표시합니다. 도움말 내용은 HTML로 표시되지만 Windows HTML 도움말 또는 Mac OS X용 Help Viewer용으로 패키징됩니다.

전체 도움말 내용은 다음과 같은 네 가지 유형의 파일로 구성됩니다. 도움말 파일에 대한 자세한 내용은 운영 체제 설명서를 참조하십시오.

- 도움말

도움말은 HTML 도움말 파일, 이미지 및 색인으로 구성됩니다. Windows에서 도움말은 확장명이 `.chm`인 파일이고, Macintosh에서 도움말은 폴더입니다.

도움말 파일은 Dreamweaver Help 폴더에 있습니다.

- help.xml 파일

help.xml 파일은 도움말 ID를 도움말 이름에 매핑합니다. 예를 들어, 다음 XML 코드는 Dreamweaver 도움말의 도움말 ID를 Windows 및 Macintosh 운영 체제의 도움말이 포함된 파일 이름에 매핑합니다.

```
<?xml version = "1.0" ?> <help-books><book-id id="DW_Using" win-mapping="UsingDreamweaver.chm" mac-mapping="Dreamweaver Help"/> </help-books>
```

각 book-id 항목에는 다음과 같은 속성이 있습니다.

- id 속성은 help.map 및 HelpDoc.js 파일에 사용된 도움말 ID입니다.
- win-mapping 속성은 Windows 도움말 이름입니다. 위 예제에서는 "UsingDreamweaver.chm"에 해당합니다.
- mac-mapping 속성은 Macintosh 도움말 이름입니다. 위 예제에서는 "Dreamweaver Help"에 해당합니다.

- help.map 파일

help.map 파일은 도움말 내용 ID를 특정 도움말에 매핑합니다. Dreamweaver는 내부적으로 도움말이 호출될 때 help.map 파일을 사용하여 특정 도움말 내용을 찾습니다.

- helpDoc.js 파일

helpDoc.js 파일을 사용하면 실제 도움말 ID와 페이지 문자열 대신 사용할 수 있는 변수 이름을 매핑할 수 있습니다.

helpDoc.js 파일은 도움말 내용 ID를 특정 도움말의 HTML 페이지에 매핑합니다. Dreamweaver는 JavaScript에서 도움말이 호출될 때 helpDoc.js 파일을 사용합니다.

인수**bookID**

- bookID 인수는 필수 항목이며 ID:page 형식으로 되어 있습니다.

ID 부분은 help.xml 파일 항목의 bookID로서, 표시할 도움말 내용을 포함하는 파일의 이름을 지정합니다. 이 항목의 page 부분은 표시할 특정 페이지를 식별합니다. 페이지는 help.map 파일에서 참조됩니다.

반환값

부울 값을 반환합니다. Dreamweaver에서 도움말 뷰어에 지정된 파일을 열 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

```
openHelpURL("DW_Using:index.htm");
```

dreamweaver.openWithApp()

지원 버전

Dreamweaver 3

설명

지정된 응용 프로그램으로 지정된 파일을 엽니다.

인수

fileURL, appURL

- fileURL 인수는 열 파일의 경로이며, file:// URL로 표시됩니다.
- appURL 인수는 파일을 열 응용 프로그램의 경로이며, file:// URL로 표시됩니다.

반환값

없음

dreamweaver.openWithBrowseDialog()

지원 버전

Dreamweaver 3

설명

[외부 편집기 선택] 대화 상자를 열어 지정된 파일을 여는 데 사용할 응용 프로그램을 사용자가 선택할 수 있도록 합니다.

인수

fileURL

- fileURL 인수는 열 파일의 경로이며, file:// URL로 표시됩니다.

반환값

없음

dreamweaver.openWithExternalTextEditor()

지원 버전

Dreamweaver 3

설명

[환경 설정] 대화 상자의 [외부 편집기] 항목에서 지정한 외부 텍스트 편집기에서 현재 문서를 엽니다.

인수

없음

반환값

없음

dreamweaver.openWithImageEditor()

지원 버전

Dreamweaver 3

설명

이름이 지정된 파일을 지정된 이미지 편집기로 엽니다.

참고: Fireworks가 이미지 편집기로 지정된 경우 이 함수는 활성 문서로 정보를 반환하는 Adobe Fireworks의 특수한 통합 메커니즘을 시작합니다. 활성 문서가 없는 경우 오류가 발생하지 않도록 하려면 [사이트] 패널에서 이 함수를 호출하지 마십시오.

인수

fileURL, appURL

- fileURL 인수는 열 파일의 경로이며, file:// URL로 표시됩니다.
- appURL 인수는 파일을 열 응용 프로그램의 경로이며, file:// URL로 표시됩니다.

반환값

없음

dreamweaver.validateFlash()

지원 버전

Dreamweaver MX

설명

로컬 컴퓨터에 Flash MX 버전 이상이 설치되어 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 로컬 컴퓨터에 Flash MX 버전 이상이 설치되어 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.insertFiles()

지원 버전

Dreamweaver CS3

설명

현재 문서의 삽입점이나 현재 선택 영역에 하나 이상의 파일을 삽입하고, 필요한 경우 사용자에게 매개 변수를 입력하도록 요청합니다.

인수

strFiles

- strFiles 인수는 삽입할 파일의 파일 경로 및 이름을 지정하는 문자열입니다. 이 함수에 여러 개의 파일 이름을 전달할 수 있습니다.

반환값

없음

dreamweaver.activateApp()

지원 버전

Dreamweaver CS3

설명

지정한 응용 프로그램을 맨 앞에 표시되는 응용 프로그램으로 지정합니다.

인수

applicationID

- applicationID는 활성화할 응용 프로그램(예: dreamweaver)을 지정하는 문자열입니다.

반환값

없음

dreamweaver.printDocument()

지원 버전

Dreamweaver CS3

설명

요청된 파일에 대해 Dreamweaver [파일] > [코드 인쇄] 명령에 해당하는 작업을 수행합니다.

인수

fileName

- fileName은 인쇄할 파일의 이름을 지정하는 문자열로, URL로 표시됩니다.

반환값
없음

dreamweaver.revealDocument()

지원 버전
Dreamweaver CS3

설명
Dreamweaver에 운영 체제 포커스를 부여하고, 지정된 파일이 Dreamweaver에서 열려 있는 경우 해당 파일을 화면 앞으로 가져옵니다.

인수
fileName

- **fileName**은 표시할 파일의 이름을 지정하는 문자열로, URL로 표시됩니다.

반환값
없음

전역 응용 프로그램 함수

전역 응용 프로그램 함수는 응용 프로그램 전체에 대해 작용합니다. 이 함수는 환경 설정 종료 및 액세스와 같은 작업을 처리합니다.

dreamweaver.beep()

지원 버전
Dreamweaver MX

설명
시스템 경고음을 생성합니다.

인수
없음

반환값
없음

예제
다음 예제에서는 dw.beep()을 호출하여 alert() 함수가 표시하는 메시지를 사용자가 주의 깊게 보도록 합니다.

```
beep(){  
    if(confirm("Is your order complete?"))  
    {  
        dreamweaver.beep();  
        alert("Click OK to submit your order");  
    }  
}
```

dreamweaver.getShowDialogsOnInsert()

지원 버전

Dreamweaver 3

설명

[환경 설정]의 [일반] 범주에 [객체 삽입시 대화 상자 보기] 옵션이 켜져 있는지 여부를 확인합니다.

인수

없음

반환값

옵션이 켜져 있는지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.quitApplication()

지원 버전

Dreamweaver 3

설명

이 함수를 호출하는 스크립트의 실행이 완료된 후 Dreamweaver를 종료합니다.

인수

없음

반환값

없음

dreamweaver.showAboutBox()

지원 버전

Dreamweaver 3

설명

[정보] 대화 상자를 엽니다.

인수

없음

반환값
없음

dreamweaver.showDynamicDataDialog()

지원 버전
Dreamweaver UltraDev 1

설명

[동적 데이터] 또는 [동적 텍스트] 대화 상자를 표시하고 사용자가 대화 상자를 닫을 때까지 기다립니다. 사용자가 [확인]을 클릭하면 showDynamicDataDialog() 함수는 사용자 문서에 삽입할 문자열을 반환합니다. 이 문자열은 데이터 소스 API 함수 generateDynamicDataRef()에서 반환되어 데이터 형식 API 함수 formatDynamicDataRef()로 전달됩니다. formatDynamicDataRef()의 반환값은 showDynamicDataDialog() 함수에서 반환된 값입니다.

인수

source, {title}

- source** 인수는 동적 데이터 객체를 나타내는 소스 코드를 포함하는 문자열입니다. 이 문자열은 이 함수에 대한 이전 호출에서 반환된 문자열과 같습니다. 이 함수는 **source** 인수의 내용을 사용하여 대화 상자의 모든 컨트롤을 초기화합니다. 따라서 사용자가 [확인]을 클릭하여 이 문자열을 만들었을 때의 상태와 똑같이 표시됩니다.

Dreamweaver는 이 문자열을 inspectDynamicDataRef() 함수로 전달하여 문자열이 트리의 노드와 일치하는지 여부를 확인합니다. 문자열이 노드와 일치하면 대화 상자가 나타날 때 해당 노드가 선택됩니다. 또한 대화 상자를 초기화하지 않는 빈 문자열을 전달할 수도 있습니다. 예를 들어, 새 항목을 만드는 데 사용되는 대화 상자는 초기화되지 않습니다.

- 선택적인 **title** 인수는 대화 상자의 제목 막대에 표시할 텍스트를 포함하는 문자열입니다. 이 인수가 제공되지 않으면 Dreamweaver는 제목 막대에 [동적 데이터]를 표시합니다.

반환값

사용자가 [확인]을 클릭하는 경우 동적 데이터 객체를 나타내는 문자열을 반환합니다.

dreamweaver.showPasteSpecialDialog()

지원 버전
Dreamweaver 8

설명

이 함수는 [선택하여 붙여넣기] 대화 상자를 표시합니다. 사용자가 [확인]을 클릭하면 showPasteSpecialDialog() 함수가 붙여넣기를 수행합니다.

인수

없음

반환값

없음

예제

```
dw.showPasteSpecialDialog();
```

dreamweaver.showPreferencesDialog()

지원 버전

Dreamweaver 3 Dreamweaver 8에 strCategory 인수가 추가되었습니다. CS4에서 업데이트되었습니다.

설명

이 함수는 [환경 설정] 대화 상자를 엽니다.

인수

{strCategory}

- 선택적인 strCategory인수는 [환경 설정] 대화 상자의 관련 범주를 여는 문자열이며, general, accessibility, "html colors"([코드 색상] 범주), "html format"([코드 포맷] 범주), "code hints", "html rewriting"([코드 다시 작성] 범주), copyPaste, "css styles", "file compare", "external editors"([파일 유형/편집기] 범주), fonts, highlighting, "invisible elements", layers, "new document", floaters([패널] 범주), browsers([브라우저에서 미리 보기] 범주), "site ftp"([사이트] 범주), "status bar", validator 중 하나여야 합니다. Dreamweaver가 유효한 창 이름으로 인수를 인식하지 않는 경우 대화 상자는 마지막 활성 창으로 열립니다. 인수가 생략된 경우에도 대화 상자가 마지막 활성 창으로 열립니다.

반환값

없음

예제

다음 예제에서는 [환경 설정] 대화 상자를 열어 [코드 색상] 범주를 선택합니다.

```
dw.showPreferencesDialog("html colors");
```

dreamweaver.showTagChooser()

지원 버전

Dreamweaver MX

설명

사용자가 [코드] 뷰로 태그를 삽입할 수 있도록 [태그 선택기] 대화 상자의 가시성을 전환합니다. 이 함수는 다른 모든 Dreamweaver 윈도우의 맨 위에 [태그 선택기] 대화 상자를 나타냅니다. 대화 상자가 표시되어 있지 않은 경우에는 이 대화 상자를 열고 맨 앞으로 가져온 다음 포커스를 설정하고, [태그 선택기]가 표시되어 있는 경우에는 대화 상자를 숨깁니다.

인수

없음

반환값

없음

dw.registerIdleHandler()

지원 버전

Dreamweaver CS3

설명

이 함수는 유휴 처리 시간 동안 정기적으로 호출되는 JavaScript 함수를 등록합니다.

인수

id, idleFunction, interval

- **id** 인수 - 등록할 유휴 작업을 식별하는 데 사용되는 고유 문자열입니다. 이 인수는 고유해야 하므로 ID 앞에 고유 식별자를 붙여야 합니다. 예를 들어, 5초에 한 번씩 경고음이 울리도록 하려는 경우, 다른 사용자가 동일한 이름으로 작업을 만들었을 수 있으므로 작업의 이름을 "beep"이라고 지정하면 안 됩니다. "acme_beep_task" 등과 같이 컨텍스트와 고유성을 모두 유지할 수 있는 이름을 사용하는 것이 좋습니다.
- **idleFunction** 인수 - 유휴 처리 시간 동안 호출되는 JavaScript 함수입니다.
- **interval** 인수는 유휴 시간 가용성에 따른 **idleFunction** 호출 시간 간격(초)입니다.

반환값

유휴 작업의 성공적인 등록 여부를 알리는 부울 값을 반환합니다.

예제

다음 예제에서는 시스템에서 5초에 한 번씩 경고음이 울리게 합니다.

```
dw.registerIdleHandler("acme_beep_task", function() { dw.beep(); }, 5);
```

dw.revokeldleHandler()

지원 버전

Dreamweaver CS3

설명

이 함수는 이전에 registerIdleHandler() 함수로 생성된 유휴 작업을 제거합니다. 이 함수를 사용하면 이전에 등록된 유휴 작업을 제거할 수 있습니다. 유휴 작업이 응용 프로그램 종료 시까지 활성 상태로 유지되어야 할 경우에는 이 함수를 호출할 필요가 없습니다. 이 경우에는 종료 직전에 유휴 작업이 자동으로 제거됩니다.

인수

id

- **id**는 제거할 등록 유휴 작업을 식별하는 데 사용되는 고유 문자열입니다. 초기에 해당 작업을 등록할 때 사용한 ID와 동일합니다.

반환값

유휴 작업의 성공적인 제거 여부를 알리는 부울 값을 반환합니다.

예제

다음 예제에서는 "dw_beep_task"라는 유휴 작업을 유휴 작업 대기열에서 제거합니다.

```
dw.revokeIdleHandler("acme_beep_task");
```

Bridge 통신 함수

Bridge 통신 함수를 사용하면 Dreamweaver와 Bridge 응용 프로그램 간에 통신할 수 있습니다. 이 통신 기능을 통해 사용자는 Bridge에서 Dreamweaver의 파일을 손쉽게 찾을 수 있습니다.

BridgeTalk.bringToFront()

지원 버전

Dreamweaver CS3

설명

BridgeTalk::bringToFront() 함수를 호출하여 지정한 응용 프로그램을 맨 앞에 표시되는 응용 프로그램으로 지정합니다.

인수

applicationID

- applicationID 인수는 활성화할 응용 프로그램을 지정하는 문자열(예: bridge 또는 dreamweaver)입니다.

반환값

없음

예제

이 예제에서는 Dreamweaver에서 browseInBridge() 함수를 구현하는 방법을 보여 줍니다. 먼저 BridgeTalk 인스턴스를 만듭니다. 그러면 가장 중요한 속성인 target 및 body가 설정됩니다. <target>은 대상 응용 프로그램으로서, 이 경우 Bridge 응용 프로그램이며 식별자는 bridge입니다. <body>는 보낼 메시지입니다. 일반적으로 <body>는 수신 후 대상 응용 프로그램이 이해하고 실행할 수 있는 스크립트입니다. send() 함수가 호출되어 <body>를 <target>으로 전송합니다.

```
if (!JSBridge.isRunning('bridge'))
{
    var bt = new BridgeTalk;
    var scriptSavePath = browsePath.replace(/["\\]/g, "%&");
    var script = "app.document.thumbnail = new Thumbnail(decodeURI('" + scriptSavePath + "'))";
    // Send the script to bridge and give it 10 sec to launch before assuming an error.
    bt.target = "bridge";
    bt.body = script;
    result = bt.send(10);
}
if (result)
    BridgeTalk.bringToFront('bridge');
```

BridgeTalk.send()

지원 버전

Dreamweaver CS3

설명

Bridge 응용 프로그램과의 통신을 설정합니다.

인수**timeout**

- The **timeout** 인수는 타임 아웃 간격을 초 단위로 설정하는 선택적 속성입니다.

반환값

Bridge 응용 프로그램과의 통신이 성공적인지 여부를 나타내는 부울 값(True = 성공, False = 실패)을 반환합니다.

예제

```
result = bridgeTalk.send(10);
```

BridgeTalk.suppressStartupScreen()

지원 버전

Dreamweaver CS3

설명

시작 후에 모달 윈도우의 숨김 여부를 결정하기 위해 -nostartupscreen 실행 옵션을 검색합니다.

반환값

시작 화면의 숨김 여부를 나타내는 부울 값을 반환합니다.

dw.browseInBridge()

지원 버전

Dreamweaver CS3

설명

Bridge에서 Dreamweaver 파일을 검색할 수 있도록 합니다. dw.browseInBridge() 함수는 Bridge 응용 프로그램을 실행합니다. Bridge가 이미 실행 중인 경우 dw.browseInBridge는 Bridge 응용 프로그램으로 전환합니다.

인수

없음

반환값

탐색 스크립트가 Bridge 응용 프로그램에 성공적으로 전송되었는지 여부를 나타내는 부울 값(true = 성공, false = 실패)을 반환합니다.

12장: 작업 영역

작업 영역 API 함수는 Adobe® Dreamweaver® 작업 영역의 요소를 만들거나 제어합니다. 이러한 함수는 다음과 같은 작업을 수행합니다.

- [작업 내역] 패널에 나타나는 단계 다시 실행
- [삽입] 막대의 객체 배치
- 키보드 함수를 사용한 탐색
- 메뉴 새로 고침
- 독립 실행형 또는 내장형 결과 윈도우 조작
- 옵션 설정
- 톨바 위치 지정
- 포커스 가져오기 또는 설정

작업 내역 함수

작업 내역 함수는 [작업 내역] 패널에 나타나는 단계의 실행 취소, 다시 실행, 기록 및 재생을 처리합니다. 여기서 단계는 문서나 문서의 선택 영역에서 반복할 수 있는 각 변경 작업을 의미합니다. `dreamweaver.historyPalette` 객체의 메서드는 현재 문서가 아니라 [작업 내역] 패널의 선택 영역을 제어하거나 선택 영역에 대해 작용합니다.

dom.redo()

지원 버전

Dreamweaver 3

설명

문서에서 가장 최근에 실행 취소한 단계를 다시 실행합니다.

인수

없음

반환값

없음

활성자

438페이지의 “[dom.canRedo\(\)](#)”를 참조하십시오.

dom.undo()

지원 버전

Dreamweaver 3

설명

문서의 이전 단계를 실행 취소합니다.

인수

없음

반환값

없음

활성자

441 페이지의 “[dom.canUndo\(\)](#)”를 참조하십시오.

dreamweaver.getRedoText()

지원 버전

Dreamweaver 3

설명

사용자가 [편집] > [다시 실행]을 선택하거나 Ctrl+Y(Windows) 또는 Command+Y(Macintosh)를 누르는 경우 다시 실행될 편집 작업과 연관된 텍스트를 가져옵니다.

인수

없음

반환값

다시 실행될 편집 작업과 연관된 텍스트를 포함하는 문자열을 반환합니다.

예제

사용자가 마지막으로 수행한 작업이 선택된 텍스트에 굵은 글꼴을 적용한 작업일 경우 dreamweaver.getRedoText() 함수를 호출하면 "반복: 굵은 글꼴 적용"이라는 문자열이 반환됩니다.

dreamweaver.getUndoText()

지원 버전

Dreamweaver 3

설명

사용자가 [편집] > [실행 취소]를 선택하거나 Ctrl+Z(Windows) 또는 Command+Z(Macintosh)를 누르는 경우 실행 취소될 편집 작업과 연관된 텍스트를 가져옵니다.

인수

없음

반환값

실행 취소될 편집 작업과 연관된 텍스트를 포함하는 문자열을 반환합니다.

예제

사용자가 마지막으로 수행한 작업이 선택된 범위의 텍스트에 CSS 스타일을 적용한 작업일 경우 `dreamweaver.getUndoText()` 함수를 호출하면 "실행 취소: 적용"이라는 문자열이 반환됩니다.

dreamweaver.playRecordedCommand()

지원 버전

Dreamweaver 3

설명

활성 문서에서 기록된 명령을 재생합니다.

인수

없음

반환값

없음

활성자

445페이지의 "[dreamweaver.canPlayRecordedCommand\(\)](#)"를 참조하십시오.

dreamweaver.redo()

지원 버전

Dreamweaver 3

설명

활성 [문서] 윈도우, 대화 상자, 부동 패널 또는 [사이트] 패널에서 가장 최근에 실행 취소한 단계를 다시 실행합니다.

인수

없음

반환값

없음

활성자

446페이지의 "[dreamweaver.canRedo\(\)](#)"를 참조하십시오.

dreamweaver.startRecording()

지원 버전

Dreamweaver 3

설명

활성 문서에서 기록 단계를 시작합니다. 이전에 기록된 명령은 즉시 삭제됩니다.

인수

없음

반환값

없음

활성자453페이지의 “[dreamweaver.isRecording\(\)](#)”을 참조하십시오(false 값을 반환해야 함).

dreamweaver.stopRecording()

지원 버전

Dreamweaver 3

설명

사용자에게 알리지 않고 기록을 중지합니다.

인수

없음

반환값

없음

활성자453페이지의 “[dreamweaver.isRecording\(\)](#)”을 참조하십시오(true 값을 반환해야 함).

dreamweaver.undo()

지원 버전

Dreamweaver 3

설명

포커스가 있는 [문서] 윈도우, 대화 상자, 부동 패널 또는 [사이트] 패널에서 이전 단계를 실행 취소합니다.

인수

없음

반환값

없음

활성자441페이지의 “[dom.canUndo\(\)](#)”를 참조하십시오.

dreamweaver.historyPalette.clearSteps()

지원 버전

Dreamweaver 3

설명

[작업 내역] 패널에서 모든 단계를 지우고 [실행 취소] 및 [다시 실행] 메뉴 항목을 비활성화합니다.

인수

없음

반환값

없음

dreamweaver.historyPalette.copySteps()

지원 버전

Dreamweaver 3

설명

지정된 작업 내역 단계를 클립보드에 복사합니다. 지정된 단계에 반복할 수 없는 작업이 포함된 경우 예기치 않게 발생할 수 있는 결과에 대한 경고가 표시됩니다.

인수

arrayOfIndices

- **arrayOfIndices** 인수는 [작업 내역] 패널의 위치 인덱스 배열입니다.

반환값

지정된 작업 내역 단계에 해당하는 JavaScript가 포함된 문자열을 반환합니다.

예제

다음 예제에서는 [작업 내역] 패널의 처음 네 단계를 복사합니다.

```
dreamweaver.historyPalette.copySteps([0,1,2,3]);
```

dreamweaver.historyPalette.getSelectedSteps()

지원 버전

Dreamweaver 3

설명

[작업 내역] 패널에서 선택된 부분을 확인합니다.

인수

없음

반환값

선택한 모든 단계의 위치 인덱스가 포함된 배열을 반환합니다. 첫 번째 위치는 0입니다.

예제

다음 그림과 같이 [작업 내역] 패널에서 두 번째, 세 번째 및 네 번째 단계가 선택된 경우 `dreamweaver.historyPalette.getSelectedSteps()` 함수를 호출하면 [1,2,3]이 반환됩니다.



dreamweaver.historyPalette.getStepCount()

지원 버전

Dreamweaver 3

설명

[작업 내역] 패널의 단계 수를 가져옵니다.

인수

없음

반환값

현재 [작업 내역] 패널에 나열된 단계 수를 나타내는 정수를 반환합니다.

dreamweaver.historyPalette.getStepsAsJavaScript()

지원 버전

Dreamweaver 3

설명

지정된 작업 내역 단계에 해당하는 JavaScript를 가져옵니다.

인수

`arrayOfIndices`

- `arrayOfIndices` 인수는 [작업 내역] 패널의 위치 인덱스 배열입니다.

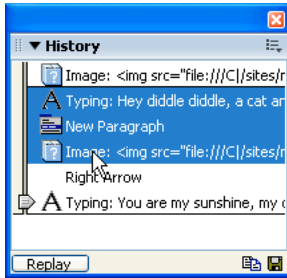
반환값

지정된 작업 내역 단계에 해당하는 JavaScript가 포함된 문자열을 반환합니다.

예제

[작업 내역] 패널에서 다음 예제에 표시된 세 단계를 선택하고

```
dreamweaver.historyPalette.getStepsAsJavaScript(dw.historyPalette.getSelectedSteps()) 함수를 호출하면
"dw.getDocumentDOM().insertText('Hey diddle diddle, a cat and a fiddle, the cow jumped over the moon.);\n
ndw.getDocumentDOM().newBlock();\n dw.getDocumentDOM().insertHTML('<img src=\n
"../wdw99/50browsers/images/sun.gif">', true);\n"이 반환됩니다.
```



dreamweaver.historyPalette.getUndoState()

지원 버전

Dreamweaver 3

설명

현재의 실행 취소 상태를 가져옵니다.

인수

없음

반환값

[작업 내역] 패널에서 [실행 취소] 표시자의 위치를 반환합니다.

dreamweaver.historyPalette.replaySteps()

지원 버전

Dreamweaver 3

설명

활성 문서에서 지정된 작업 내역 단계를 재실행합니다. 지정된 단계에 반복할 수 없는 작업이 포함된 경우 예기치 않게 발생할 수 있는 결과에 대한 경고가 표시됩니다.

인수

arrayOfIndices

- **arrayOfIndices** 인수는 [작업 내역] 패널의 위치 인덱스 배열입니다.

반환값

지정된 작업 내역 단계에 해당하는 JavaScript가 포함된 문자열을 반환합니다.

예제

`dreamweaver.historyPalette.replaySteps([0,2,3])` 함수를 호출하면 [작업 내역] 패널에서 첫 번째, 세 번째 및 네 번째 단계가 실행됩니다.

dreamweaver.historyPalette.saveAsCommand()

지원 버전

Dreamweaver 3

설명

사용자가 지정된 단계를 명령으로 저장할 수 있도록 [명령으로 저장] 대화 상자를 엽니다. 단계에 반복할 수 없는 작업이 포함된 경우 예기치 않게 발생할 수 있는 결과에 대한 경고가 표시됩니다.

인수

arrayOfIndices

- **arrayOfIndices** 인수는 [작업 내역] 패널의 위치 인덱스 배열입니다.

반환값

지정된 작업 내역 단계에 해당하는 JavaScript가 포함된 문자열을 반환합니다.

예제

다음 예제에서는 [작업 내역] 패널의 네 번째, 여섯 번째 및 여덟 번째 단계를 명령으로 저장합니다.

```
dreamweaver.historyPalette.saveAsCommand([3,5,7]);
```

dreamweaver.historyPalette.setSelectedSteps()

지원 버전

Dreamweaver 3

설명

[작업 내역] 패널에서 지정된 단계를 선택합니다.

인수

arrayOfIndices

- **arrayOfIndices** 함수는 [작업 내역] 패널의 위치 인덱스 배열입니다. 인수를 제공하지 않으면 모든 단계가 선택 취소됩니다.

반환값

없음

예제

다음 예제에서는 [작업 내역] 패널에서 첫 번째, 두 번째 및 세 번째 단계를 선택합니다.

```
dreamweaver.historyPalette.setSelectedSteps([0,1,2]);
```

dreamweaver.historyPalette.setUndoState()

지원 버전

Dreamweaver 3

설명

지정된 실행 취소 상태에 이르기까지 필요한 횟수만큼 실행 취소 또는 다시 실행 작업을 수행합니다.

인수

undoState

- **undoState** 인수는 `dreamweaver.historyPalette.getUndoState()` 함수가 반환하는 객체입니다.

반환값

없음

객체 삽입 함수

객체 삽입 함수는 [삽입] 막대에 있는 객체 또는 [삽입] 메뉴에 나열된 객체와 연관된 작업을 처리합니다.

dreamweaver.objectPalette.getMenuDefault()

지원 버전

Dreamweaver MX 2004

설명

연관된 메뉴에 대한 기본 항목의 ID 문자열을 검색합니다.

인수

menuId

- **menuId** 인수는 `insertbar.xml` 파일에 메뉴를 정의하는 문자열입니다.

반환값

기본 항목의 ID를 정의하는 문자열 값을 반환합니다.

예제

다음 예제에서는 [미디어] 메뉴의 현재 기본 객체를 `defId` 변수에 할당합니다.

```
var defId = dw.objectPalette.getMenuDefault("DW_Media");
```

dreamweaver.objectPalette.setMenuDefault()

지원 버전

Dreamweaver MX 2004

설명

팝업 메뉴의 기본 객체를 설정합니다. 기본 객체의 아이콘은 삽입 막대의 지정된 팝업 메뉴를 나타냅니다. 사용자는 기본 객체를 클릭하여 삽입하거나 기본 객체 옆에 있는 화살표를 클릭하여 팝업 메뉴를 열고 해당 메뉴의 다른 객체를 볼 수 있습니다. 다음에 Dreamweaver를 열거나 [Extensions 새로 고침] 명령을 사용하면 새 메뉴가 기본값으로 설정됩니다.

인수

menuId, defaultId

- **menuId** 인수는 insertbar.xml 파일에 메뉴를 정의하는 문자열입니다.
- **defaultId** 인수는 insertbar.xml 필드에 새 기본 객체를 정의하는 문자열입니다.

반환값

부울 값을 반환합니다. 새 기본값이 성공적으로 설정되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

다음 예제에서는 Flash 객체를 [미디어] 메뉴의 기본 객체로 설정합니다.

```
dw.objectPalette.setMenuDefault("DW_Media", "DW_Flash");
```

dreamweaver.reloadObjects()

지원 버전

Dreamweaver MX 2004

설명

[삽입] 막대의 모든 객체를 새로 고칩니다. 이 함수를 실행하면 Ctrl 키를 누른 상태에서 [삽입] 막대의 [범주] 메뉴를 클릭한 다음 [Extensions 새로 고침] 메뉴 옵션을 선택하는 것과 같은 결과가 나타납니다.

인수

없음

반환값

부울 값을 반환합니다. 객체가 성공적으로 로드되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.convertActiveContent()

지원 버전

Dreamweaver CS3

설명

지정된 문서에서 모든 액티브 콘텐츠를 변환합니다.

인수

forceUpdate

- **forceUpdate** 인수는 사용자의 환경 설정을 무시할지 여부를 나타내는 부울 값입니다(true = 무시). 이 인수는 선택 사항입니다.

반환값

부울 값을 반환합니다. 모든 액티브 콘텐츠가 성공적으로 변환되면 **true**를 반환합니다. 템플릿 인스턴스의 잠긴 영역에 있는 객체 태그와 같이 변환해야 하는 일부 액티브 콘텐츠가 변환되지 않았으면 **false**를 반환합니다.

예제

```
if( !dom.convertActiveContent(true) ) {
    alert(dw.loadString("ActiveContent/notAllConverted"));
}
```

dom.convertNextActiveContent()**지원 버전**

Dreamweaver CS3

설명

현재 편집(한 단계의 실행 취소가 가능한 작업)의 나머지 부분에 대해 삽입되는 다음 객체 태그의 스크립트가 만들어지도록 지정합니다. 이 함수를 사용하면 타사 확장 기능을 사용하여 특정 액티브 콘텐츠에 적합한 스크립트를 생성할 수 있습니다.

인수

없음

반환값

없음

예제

```
dom.convertNextActiveContent();
dom.insertHTML("<object classid=\"clsid:D27CDB6E-AE6D-11cf-96B8-444553540000\" codebase=\"
http://download.Macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=7,0,19,0\"
width=\"100\" height=\"22\"><param name=\"movie\" value=\"button1.swf\" /><param name=\"
quality\" value=\"high\" /><embed src=\"button1.swf\" quality=\"high\" pluginspage=\"
http://www.Macromedia.com/go/getflashplayer\" type=\"application/
x-shockwave-flash\" width=\"100\" height=\"22\"></embed></object>");
```

키보드 함수

키보드 함수를 사용하면 화살표, 백스페이스, Delete, Page Up 및 Page Down 키를 누를 때처럼 문서 내비게이션 작업을 수행할 수 있습니다. arrowLeft() 및 backspaceKey()와 같은 일반적인 화살표 및 키 함수 외에도 다음 또는 이전 단어나 단락으로 이동하는 방법과 행이나 문서의 처음 또는 행이나 문서의 끝으로 이동하는 방법도 제공됩니다.

dom.arrowDown()**지원 버전**

Dreamweaver 3

설명

지정한 횟수만큼 삽입점을 아래로 이동합니다.

인수**{nTimes}, {bShiftIsDown}**

- **nTimes** 인수는 삽입점이 아래로 이동하는 횟수입니다. 이 인수를 생략하면 기본값은 1입니다.
- **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false입니다.

반환값

없음

dom.arrowLeft()**지원 버전**

Dreamweaver 3

설명

지정한 횟수만큼 삽입점을 왼쪽으로 이동합니다.

인수**{nTimes}, {bShiftIsDown}**

- 선택적인 **nTimes** 인수는 삽입점이 왼쪽으로 이동하는 횟수입니다. 이 인수를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false입니다.

반환값

없음

dom.arrowRight()**지원 버전**

Dreamweaver 3

설명

지정한 횟수만큼 삽입점을 오른쪽으로 이동합니다.

인수**{nTimes}, {bShiftIsDown}**

- 선택적인 **nTimes** 인수는 삽입점이 오른쪽으로 이동하는 횟수입니다. 이 인수를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false입니다.

반환값

없음

dom.arrowUp()

지원 버전

Dreamweaver 3

설명

이 함수는 지정한 횟수만큼 삽입점을 위로 이동합니다.

인수

{nTimes}, {bShiftIsDown}

- 선택적인 **nTimes** 인수는 삽입점이 위로 이동하는 횟수입니다. 이 인수를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 **false**입니다.

반환값

없음

dom.backspaceKey()

지원 버전

Dreamweaver 3

설명

이 함수를 실행하면 지정한 횟수만큼 백스페이스 키를 누르는 것과 동일한 결과가 나타납니다. 정확한 동작은 현재 선택 영역이 있는지, 아니면 삽입점만 있는지에 따라 다릅니다.

인수

{nTimes}

- 선택적인 **nTimes** 인수는 백스페이스 작업이 발생하는 횟수입니다. 이 인수를 생략하면 기본값은 1입니다.

반환값

없음

dom.deleteKey()

지원 버전

Dreamweaver 3

설명

이 함수를 실행하면 지정한 횟수만큼 **Delete** 키를 누르는 것과 동일한 결과가 나타납니다. 정확한 동작은 현재 선택 영역이 있는지, 아니면 삽입점만 있는지에 따라 다릅니다.

인수**{nTimes}**

- 선택적인 **nTimes** 인수는 Delete 작업이 발생하는 횟수입니다. 이 인수를 생략하면 기본값은 1입니다.

반환값

없음

dom.endOfDocument()**지원 버전**

Dreamweaver 3

설명

삽입점을 문서의 끝으로 이동합니다. 즉, 포커스가 있는 윈도우에 따라 [문서] 윈도우에서는 마지막으로 보이는 내용 다음으로 삽입점을 이동하고, 코드 관리자에서는 닫기 HTML 태그 다음으로 이동합니다.

인수**{bShiftIsDown}**

- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false 입니다.

반환값

없음

dom.endOfLine()**지원 버전**

Dreamweaver 3

설명

삽입점을 행의 끝으로 이동합니다.

인수**{bShiftIsDown}**

- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false 입니다.

반환값

없음

dom.nextParagraph()**지원 버전**

Dreamweaver 3

설명

삽입점을 다음 단락의 처음으로 이동하거나, **nTimes**가 1보다 큰 경우에는 여러 단락을 건너뛵니다.

인수

{**nTimes**}, {**bShiftIsDown**}

- 선택적인 **nTimes** 인수는 삽입점이 앞으로 이동하는 단락 수입니다. 이 인수를 생략하면 기본값은 1입니다.
- **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false입니다.

반환값

없음

dom.nextWord()

지원 버전

Dreamweaver 3

설명

삽입점을 다음 단어의 처음으로 이동하거나, **nTimes**가 1보다 큰 경우에는 여러 단어를 건너뛵니다.

인수

{**nTimes**}, {**bShiftIsDown**}

- 선택적인 **nTimes** 인수는 삽입점이 앞으로 이동하는 단어 수입니다. 이 인수를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false입니다.

반환값

없음

dom.pageDown()

지원 버전

Dreamweaver 3

설명

삽입점을 한 페이지 아래로 이동합니다(Page Down 키를 누르는 것과 동일함).

인수

{**nTimes**}, {**bShiftIsDown**}

- 선택적인 **nTimes** 인수는 삽입점이 아래로 이동하는 페이지 수입니다. 이 인수를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false입니다.

반환값

없음

dom.pageUp()

지원 버전

Dreamweaver 3

설명

삽입점을 한 페이지 위로 이동합니다(Page Up 키를 누르는 것과 동일함).

인수

{nTimes}, {bShiftIsDown}

- 선택적인 **nTimes** 인수는 삽입점이 위로 이동하는 페이지 수입니다. 이 인수를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 **false**입니다.

반환값

없음

dom.previousParagraph()

지원 버전

Dreamweaver 3

설명

삽입점을 이전 단락의 처음으로 이동하거나, **nTimes**가 1보다 큰 경우에는 여러 단락을 건너뛵니다.

인수

{nTimes}, {bShiftIsDown}

- 선택적인 **nTimes** 인수는 삽입점이 뒤로 이동하는 단락 수입니다. 이 인수를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 **false**입니다.

반환값

없음

dom.previousWord()

지원 버전

Dreamweaver 3

설명

삽입점을 이전 단어의 처음으로 이동하거나, **nTimes**가 1보다 큰 경우에는 여러 단어를 건너뛵니다.

인수**{nTimes}, {bShiftIsDown}**

- 선택적인 **nTimes** 인수는 삽입점이 뒤로 이동하는 단어 수입니다. 이 인수를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false입니다.

반환값

없음

dom.startOfDocument()**지원 버전**

Dreamweaver 3

설명

삽입 포인터를 문서의 처음으로 이동합니다. 즉, 포커스가 있는 윈도우에 따라 [문서] 윈도우에서는 첫 번째 보이는 내용 앞으로 삽입점을 이동하고, 코드 관리자에서는 열기 HTML 태그 앞으로 이동합니다.

인수**{bShiftIsDown}**

- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false입니다.

반환값

없음

dom.startOfLine()**지원 버전**

Dreamweaver 3

설명

삽입점을 행의 처음으로 이동합니다.

인수**{bShiftIsDown}**

- 선택적인 **bShiftIsDown** 인수는 선택 영역을 확장할지 여부를 나타내는 부울 값입니다. 이 인수를 생략하면 기본값은 false입니다.

반환값

없음

dreamweaver.mapKeyCodeToChar()

지원 버전

Dreamweaver 4

설명

이벤트 객체의 `keyCode` 필드에서 가져온 키 코드를 문자로 변환합니다. 키 코드가 HOME, PGUP 등의 특수 키인지 여부를 확인해야 합니다. 키 코드가 특수 키가 아닌 경우 이 메서드를 사용하여 키 코드를 사용자에게 표시할 수 있는 문자 코드로 변환할 수 있습니다.

인수

`keyCode`

- `keyCode` 인수는 문자로 변환할 키 코드입니다.

반환값

매핑이 성공하면 문자 코드를 반환하고 그렇지 않으면 0을 반환합니다.

메뉴 함수

메뉴 함수는 Dreamweaver에서 메뉴의 최적화 및 새로 고침을 처리합니다. `dreamweaver.getMenuNeedsUpdating()` 함수와 `dreamweaver.notifyMenuUpdated()` 함수는 Dreamweaver에 내장된 동적 메뉴가 불필요하게 업데이트되지 않도록 하기 위해 특별히 설계되었습니다. 자세한 내용은 134페이지의 “[dreamweaver.getMenuNeedsUpdating\(\)](#)” 및 135페이지의 “[dreamweaver.notifyMenuUpdated\(\)](#)”를 참조하십시오.

dreamweaver.getMenuNeedsUpdating()

지원 버전

Dreamweaver 3

설명

지정된 메뉴를 업데이트해야 하는지 확인합니다.

인수

`menuId`

- `menuId` 인수는 `menus.xml` 파일에 지정된 메뉴 항목의 `id` 속성 값을 포함하는 문자열입니다.

반환값

메뉴를 업데이트해야 하는지 여부를 나타내는 부울 값을 반환합니다. 이 함수는 `dreamweaver.notifyMenuUpdated()`가 이 `menuId`와 함께 호출된 후 `menuListFunction`의 반환값이 변경되지 않은 경우에만 `false`를 반환합니다. 자세한 내용은 135페이지의 “[dreamweaver.notifyMenuUpdated\(\)](#)”를 참조하십시오.

dreamweaver.notifyMenuUpdated()

지원 버전

Dreamweaver 3

설명

지정된 메뉴를 업데이트해야 할 때 Dreamweaver에 알립니다.

인수

menuId, menuListFunction

- **menuId** 인수는 menus.xml 파일에 지정된 메뉴 항목의 id 속성 값을 포함하는 문자열입니다.
- **siteOrURL** 인수는 "dw.cssStylePalette.getStyles()", "dw.getDocumentDOM().getFrameNames()", "dw.getDocumentDOM().getEditableRegionList()", "dw.getBrowserList()", "dw.getRecentFileList()", "dw.getTranslatorList()", "dw.getFontList()", "dw.getDocumentList()", "dw.htmlStylePalette.getStyles()" 또는 "site.getSites()" 문자열 중 하나여야 합니다.

반환값

없음

dreamweaver.reloadMenus()

지원 버전

Dreamweaver 3

설명

Configuration 폴더의 menus.xml 파일에서 전체 메뉴 구조를 새로 고칩니다.

인수

없음

반환값

없음

결과 윈도우 함수

결과 윈도우 함수를 사용하면 [결과] 패널 그룹의 내장 패널과 상호 작용하거나 서식이 적용된 데이터 열을 표시하는 독립 실행형 윈도우를 만들 수 있습니다.

내장 결과 패널 그룹

이러한 함수는 [결과] 패널 그룹에 출력을 생성합니다. [결과] 패널 그룹에는 검색, 소스 유효성 검사, 사이트 전체 보고서, 브라우저 호환성 확인, 서버 디버깅, FTP 로깅 및 링크 확인 등의 탭 패널이 표시됩니다.

특정 자식 패널

다음 자식 패널은 Dreamweaver 인터페이스에 항상 존재하는 내장형 [결과] 윈도우이며 직접 액세스할 수 있습니다.

- `dreamweaver.resultsPalette.siteReports`
- `dreamweaver.resultsPalette.validator`
- `dreamweaver.resultsPalette.bcc`

이러한 패널은 [결과] 윈도우이므로 독립 실행형 [결과] 윈도우에 대해 정의된 다음 메서드를 사용할 수 있습니다.

- `getItem()`
- `getItemCount()`
- `getSelectedItem()`
- `setSelectedItem()`

`resWin` 메서드를 사용하는 방법에 대한 자세한 내용은 140페이지의 “[독립 실행형 결과 윈도우](#)”를 참조하십시오.

활성 자식 패널

다음 일반 API 함수는 활성 상태의 모든 자식 패널에 적용됩니다. 일부 자식 패널은 이러한 함수 중 일부를 무시할 수도 있습니다. 활성 자식 패널이 지원하지 않는 함수를 호출하면 아무 영향을 주지 않습니다.

dreamweaver.showResults()

지원 버전

Dreamweaver MX 2004

설명

지정된 결과 부동 패널을 열고 항목을 선택합니다.

참고: 이 함수는 [결과] 패널 그룹의 [유효성 검사], [브라우저 호환성 확인] 및 [사이트 보고서] 패널에서만 지원됩니다.

인수

floaterName, floaterIndex

- **floaterName** 인수는 열리는 결과 부동 패널을 지정하는 문자열입니다. 유효한 값은 'validation' 또는 'reports'입니다.
- **floaterIndex** 인수는 숫자 또는 문자열입니다. [결과] 패널에서 선택할 항목의 인덱스를 지정하려면 숫자를 사용하고, 문서의 URL을 지정하려면 문자열을 사용합니다. URL을 지정하는 경우 함수는 해당 문서에 표시되는 첫 번째 항목을 선택합니다.

반환값

없음

예제

다음 예제에서는 문서에서 현재 선택된 영역의 오프셋에 오류가 있는지 확인한 다음 오류가 있으면 [결과] 패널의 지정된 윈도우(`floaterName`)에 오류를 표시합니다. 오류가 없으면 [결과] 패널의 [브라우저 호환성 확인] 윈도우를 열고 현재 문서에 표시되는 첫 번째 항목을 표시합니다.

```

var offset = dw.getDocumentDOM().source.getSelection()[0];
var errors = dw.getDocumentDOM().source.getValidationErrorsForOffset(offset);
if ( errors && errors.length > 0 )
    dw.showResults( errors[0].floaterName, errors[0].floaterIndex );
else
    dw.showResults( 'bcc', dw.getDocumentDOM().URL );

```

dreamweaver.resultsPalette.siteReports.addResultItem()

지원 버전

Dreamweaver 4

설명

processfile() 함수가 처리하는 파일의 정보에 따라 [사이트 보고서] 패널에 새 결과 항목을 추가합니다.

이 함수는 사이트 보고서의 processFile() 콜백에서만 사용할 수 있습니다. 사이트 보고서에 대한 자세한 내용은 **Dreamweaver** 확장에서 "보고서"를 참조하십시오.

인수

strFilePath, strIcon, strDisplay, strDesc, {iLineNo}, {iStartSel}, {iEndSel}

- **strFilePath** 인수는 처리할 파일의 전체 URL 경로입니다.
- **strIcon** 인수는 사용할 아이콘의 경로입니다. 내장 아이콘을 표시하려면 아이콘의 전체 경로 대신 "1"부터 "10"까지의 값을 사용하고, 아이콘을 표시하지 않으려면 "0"을 사용합니다. 다음 표에서는 "1"부터 "10"까지의 값에 해당하는 아이콘을 보여 줍니다.

1 2 3 4 5 6 7 8 9 10


- **strDisplay** 인수는 결과 윈도우의 첫 번째 열(대개 파일 이름)에 표시할 문자열입니다.
- **strDesc** 인수는 항목과 함께 제공되는 설명입니다.
- **iLineNo** 인수는 파일의 행 수입니다(선택 항목).
- **iStartSel** 인수는 파일로의 오프셋 시작을 나타냅니다. 이 인수는 선택 사항이며 **iEndSel** 인수와 함께 사용해야 합니다.
- **iEndSel** 인수는 파일로의 오프셋 끝을 나타냅니다. **iStartSel**을 사용하는 경우에는 반드시 사용해야 합니다.

반환값

없음

dreamweaver.resultsPalette.clear()

지원 버전

Dreamweaver MX

설명

포커스가 있는 패널의 내용을 지웁니다.

인수

없음

반환값

없음

활성자

453페이지의 “[dreamweaver.resultsPalette.canClear\(\)](#)”를 참조하십시오.

dreamweaver.resultsPalette.Copy()

지원 버전

Dreamweaver MX

설명

복사한 메시지를 포커스가 있는 윈도우로 보냅니다. FTP 로깅 윈도우에 많이 사용됩니다.

인수

없음

반환값

없음

활성자

454페이지의 “[dreamweaver.resultsPalette.canCopy\(\)](#)”를 참조하십시오.

dreamweaver.resultsPalette.cut()

지원 버전

Dreamweaver MX

설명

잘라낸 메시지를 포커스가 있는 윈도우로 보냅니다. FTP 로깅 윈도우에 많이 사용됩니다.

인수

없음

반환값

없음

활성자

454페이지의 “[dreamweaver.resultsPalette.canCut\(\)](#)”을 참조하십시오.

dreamweaver.resultsPalette.Paste()

지원 버전

Dreamweaver MX

설명

붙여 넣은 메시지를 포커스가 있는 윈도우로 보냅니다. FTP 로깅 윈도우에 많이 사용됩니다.

인수

없음

반환값

없음

활성자

454페이지의 “[dreamweaver.resultsPalette.canPaste\(\)](#)”를 참조하십시오.

dreamweaver.resultsPalette.openInBrowser

지원 버전

Dreamweaver MX

설명

기본 브라우저에 보고서(사이트 보고서, 브라우저 대상 확인, 유효성 검사 및 링크 확인기)를 전송합니다.

인수

없음

반환값

없음

활성자

455페이지의 “[dreamweaver.resultsPalette.canOpenInBrowser\(\)](#)”를 참조하십시오.

dreamweaver.resultsPalette.openInEditor()

지원 버전

Dreamweaver MX

설명

특정 보고서(사이트 보고서, 브라우저 대상 확인, 유효성 검사 및 링크 확인기)의 선택한 행으로 이동하고 편집기에서 해당 문서를 엽니다.

인수

없음

반환값

없음

활성자

455페이지의 “[dreamweaver.resultsPalette.canOpenInEditor\(\)](#)”를 참조하십시오.

dreamweaver.resultsPalette.save()

지원 버전

Dreamweaver MX

설명

save() 함수를 지원하는 윈도우(사이트 보고서, 브라우저 대상 확인, 유효성 검사 및 링크 확인기)에서 [저장] 대화 상자를 엽니다.

인수

없음

반환값

없음

활성자

455페이지의 “[dreamweaver.resultsPalette.canSave\(\)](#)”를 참조하십시오.

dreamweaver.resultsPalette.selectAll()

지원 버전

Dreamweaver MX

설명

[모두 선택] 명령을 포커스가 있는 윈도우로 보냅니다.

인수

없음

반환값

없음

활성자

456페이지의 “[dreamweaver.resultsPalette.canSelectAll\(\)](#)”을 참조하십시오.

독립 실행형 결과 윈도우

dreamweaver.createResultsWindow() 함수는 결과 윈도우를 만듭니다.

dreamweaver.createResultsWindow()

지원 버전

Dreamweaver 4

설명

새로운 [결과] 윈도우를 만들고 이 윈도우에 JavaScript 객체 참조를 반환합니다.

인수

strName, arrColumns

- **strName** 인수는 윈도우의 제목에 사용할 문자열입니다.

- **arrColumns** 인수는 목록 컨트롤에서 사용할 열 이름의 배열입니다.

반환값

생성된 윈도우에 대한 객체 참조를 반환합니다.

resWin.addItem()

지원 버전

Dreamweaver 4

설명

[결과] 윈도우에 새 항목을 추가합니다.

참고: 140페이지의 “[dreamweaver.createResultsWindow\(\)](#)”로 만든 독립 실행형 결과 윈도우만 사용합니다. [유효성 검사], [브라우저 호환성 확인] 또는 [사이트 보고서]를 비롯한 내장형 결과 윈도우에는 `resWin.addItem()` 함수를 사용할 수 없습니다.

인수

resultWindowObj, strIcon, strDesc, itemData, iStartSel, iEndSel, colNdata

- **resultWindowObj** 인수는 `createResultsWindow()` 함수가 반환하는 객체입니다.
- **strIcon** 인수는 사용할 아이콘에 대한 경로를 지정하는 문자열입니다. 내장 아이콘을 표시하려면 아이콘의 전체 경로 대신 "1"부터 "10"까지의 값을 사용하고, 아이콘이 없으면 "0"을 지정합니다. 다음 표에서는 "1"부터 "10"까지의 값에 해당하는 아이콘을 보여 줍니다.

1 2 3 4 5 6 7 8 9 10


- **strDesc** 인수는 항목에 대한 자세한 설명입니다. 설명이 없으면 "0"을 지정합니다.
- **itemData** 인수는 문서 행 번호 등의 추가할 항목에 대한 특정 데이터를 저장하는 데 사용할 수 있는 문자열입니다.
- **iStartSel** 인수는 파일의 선택 영역 오프셋 시작입니다. 오프셋을 지정하지 않으려면 `null` 값을 지정합니다.
- **iEndSel** 인수는 파일의 선택 영역 오프셋 끝입니다. 오프셋을 지정하지 않으려면 `null` 값을 지정합니다.
- **colNdata** 인수는 각 열에 대한 데이터를 제공하는 문자열의 배열입니다. 즉, 열이 세 개 있으면 세 개의 문자열로 이루어진 배열입니다.

반환값

부울 값을 반환합니다. 항목이 성공적으로 추가되면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

예제

다음 예제에서는 Frodo, Sam 및 Gollum이라는 열 머리글이 있는 `resWin`이라는 [결과] 윈도우를 만듭니다. `resWin.addItem()` 함수를 호출하면 폴더 아이콘이 추가된 다음 세 문자열 `msg1`, `msg2` 및 `msg3`이 윈도우에 대해 정의된 세 열에 추가됩니다.

```
var resWin = dw.createResultsWindow("Test Window", ["Frodo", "Sam", "Gollum"]);
resWin.addItem(resWin, "3", "Description", null, null, null, ["msg1", "msg2", "msg3"]);
```

resWin.getItem()

지원 버전

Dreamweaver 4

설명

항목을 추가한 명령 이름과 addItem() 함수에 전달된 동일한 문자열을 포함하는 항목 배열을 가져옵니다.

인수

itemIndex

- itemIndex 인수는 데이터를 가져온 항목의 인덱스입니다.

반환값

문자열 배열을 반환합니다. 이 배열의 첫 번째 요소는 항목에 추가된 명령 이름이고 나머지 요소는 addItem() 함수에 전달된 동일한 문자열입니다.

resWin.getItemCount()

지원 버전

Dreamweaver 4

설명

목록에 있는 항목 수를 가져옵니다.

인수

없음

반환값

목록에 있는 항목 수를 반환합니다.

resWin.getSelectedItem()

지원 버전

Dreamweaver 4

설명

선택한 항목의 인덱스를 가져옵니다.

인수

없음

반환값

현재 선택한 항목의 인덱스를 반환합니다.

resWin.setButtons()

지원 버전

Dreamweaver 4

설명

arrButtons 인수에서 지정한 버튼을 설정합니다.

인수

cmdDoc, **arrButtons**

- **cmdDoc** 인수는 함수를 호출하는 명령을 나타내는 문서 객체입니다. 명령은 **this** 키워드를 사용해야 합니다.
- **arrButtons** 인수는 버튼을 클릭할 때 실행할 버튼 텍스트와 JavaScript 코드에 해당하는 문자열의 배열입니다. 이것은 **commandButtons()** 함수가 명령에 대해 작동하는 방식과 유사합니다. 윈도우에서는 두 개의 버튼만 설정할 수 있습니다.

반환값

없음

resWin.setCallbackCommands()

지원 버전

Dreamweaver 4

설명

processFile() 메서드를 호출할 명령을 [결과] 윈도우에 알립니다. 이 함수가 호출되지 않으면 [결과] 윈도우를 생성한 명령이 호출됩니다.

인수

arrCmdNames

- **arrCmdNames** 인수는 **processFile()** 함수를 호출할 명령 이름의 배열입니다.

반환값

없음

resWin.setColumnWidths()

지원 버전

Dreamweaver 4

설명

각 열의 폭을 설정합니다.

인수

arrWidth

- **arrWidth** 인수는 컨트롤의 각 열에 사용할 폭을 나타내는 정수의 배열입니다.

반환값
없음

resWin.setFileList()

지원 버전
Dreamweaver 4

설명
[결과] 윈도우에 파일 목록이나 폴더 목록 또는 둘 다를 제공하여 처리할 명령 세트를 호출합니다.

인수
arrFilePaths, bRecursive

- **arrFilePaths** 인수는 반복 적용할 파일 또는 폴더 경로의 배열입니다.
- **bRecursive** 인수는 반복 적용이 재귀적인지(true) 또는 아닌지(false)를 나타내는 부울 값입니다.

반환값
없음

resWin.setSelectedItem()

지원 버전
Dreamweaver 4

설명
선택한 항목을 **itemIndex**로 지정된 값으로 설정합니다.

인수
itemIndex

- 선택할 목록에 있는 항목의 인덱스입니다.

반환값
이전에 선택한 항목의 인덱스를 반환합니다.

resWin.setTitle()

지원 버전
Dreamweaver 4

설명
윈도우의 제목을 설정합니다.

인수**strTitle**

- **strTitle** 인수는 부동 패널의 새 이름입니다.

반환값

없음

resWin.startProcessing()**지원 버전**

Dreamweaver 4

설명

파일 처리를 시작합니다.

인수

없음

반환값

없음

resWin.stopProcessing()**지원 버전**

Dreamweaver 4

설명

파일 처리를 중단합니다.

인수

없음

반환값

없음

서버 디버깅

Dreamweaver에서는 Adobe ColdFusion에서 파일을 요청하고 포함된 브라우저에 응답을 표시할 수 있습니다. 서버에서 응답이 반환될 때 Dreamweaver에서는 알려진 서명을 포함하는 XML 패키지를 응답에서 검색합니다. 해당 서명을 포함하는 XML을 발견할 경우 해당 XML을 처리하고 포함된 정보를 트리 컨트롤에 표시합니다. 이 트리는 다음 항목에 대한 정보를 표시합니다.

- 렌더링된 CFM 페이지를 생성하는 데 사용되는 모든 템플릿, 사용자 정의 태그 및 포함 파일
- 예외
- SQL 쿼리
- 객체 쿼리

- 변수
- 서버 디버그 정보 추적

또한 [서버 디버그] 패널에는 다른 서버 모델의 디버그 데이터가 표시될 수 있습니다. Dreamweaver에서 다른 서버 모델을 디버그하도록 설정하려면 `dreamweaver.resultsPalette.debugWindow.addDebugContextData()` 함수를 사용합니다.

dreamweaver.resultsPalette.debugWindow.addDebugContextData()

지원 버전

Dreamweaver MX

설명

[사이트 정의] 대화 상자에 지정된 서버에서 반환되는 사용자 정의된 XML 파일을 해석합니다. XML 파일의 내용은 [서버 디버그] 패널에서 트리 데이터로 표시되므로 [서버 디버그] 패널을 사용하여 다양한 서버 모델에서 생성된 내용을 평가할 수 있습니다.

인수

treedata

- **treedata** 인수는 서버에서 반환하는 XML 문자열입니다. XML 문자열은 다음 서식을 사용해야 합니다.

server debug node	디버그 XML 데이터의 루트 노드
debugnode	모든 노드에 해당
context	컨텍스트 목록에 나타나는 항목의 이름
icon	트리 노드에 사용할 아이콘
name	표시할 이름
value	표시할 값
timestamp	컨텍스트 노드에만 적용 가능

다음은 선택적으로 사용되는 문자열입니다.

jumptoline	특정 행 번호에 연결
template	URL의 일부인 템플릿 파일의 이름
path	서버측에서 본 파일 경로
line number	파일 내의 행 번호
start position	행 내의 열기 문자 오프셋
end position	행 내의 끝 문자 오프셋

예:

```
<serverdebuginfo>
  <context>
    <template><![CDATA[/ooo/master.cfm]]></template>
    <path><![CDATA[C:\server\wwwroot\ooo\master.cfm]]></path>
    <timestamp><![CDATA[0:0:0.0]]></timestamp>
  </context>
  <debugnode>
    <name><![CDATA[CGI]]></name>
    <icon><![CDATA[ServerDebugOutput/ColdFusion/CGIVariables.gif]]></icon>
    <debugnode>
      <name><![CDATA[Pubs.name.sourceURL]]></name>
      <icon><![CDATA[ServerDebugOutput/ColdFusion/Variable.gif]]></icon>
      <value><![CDATA[jdbc:Macromedia:sqlserver:
        //name.Macromedia.com:1111;databaseName=Pubs]]></value>
    </debugnode>
  </debugnode>
  <debugnode>
    <name><![CDATA[Element Snippet is undefined in class
coldfusion.compiler.TagInfoNotFoundException]]></name>
    <icon><![CDATA[ServerDebugOutput/ColdFusion/Exception.gif]]></icon>
    <jumptoline lineNumber="3" startposition="2" endposition="20">
      <template><![CDATA[/ooo/master.cfm]]></template>
      <path><![CDATA[C:\Neo\wwwroot\ooo\master.cfm]]></path>
    </jumptoline>
  </debugnode>
</serverdebuginfo>
```

반환값

없음

전환 함수

전환 함수는 다양한 옵션을 가져와서 설정하거나 해제합니다.

dom.getEditNoFramesContent()

지원 버전

Dreamweaver 3

설명

이 함수는 [수정] > [프레임세트] > [비프레임 내용 편집] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. NOFRAMES 내용이 활성 뷰이면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getHideAllVisualAids()

지원 버전

Dreamweaver 4

설명

이 함수는 시각 도구가 숨김으로 설정되어 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [시각 도구 모두 숨김]이 숨김으로 설정되어 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getPreventLayerOverlaps()

지원 버전

Dreamweaver 3

설명

이 함수는 [레이어 겹침 방지] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. [레이어 겹침 방지] 옵션이 켜져 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowAutoIndent()

지원 버전

Dreamweaver 4

설명

이 함수는 문서 윈도우의 [코드] 뷰에 자동 들여쓰기 기능이 켜져 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 자동 들여쓰기 기능이 켜져 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowFrameBorders()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [프레임 테두리] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. 프레임 테두리가 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowGrid()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [격자] > [격자 표시] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. 격자가 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowHeaderView()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [헤드 내용] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. 헤드 내용이 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowInvalidHTML()

지원 버전

Dreamweaver 4

설명

이 함수는 현재 문서 윈도우의 [코드] 뷰에 잘못된 HTML 코드가 강조 표시되어 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 잘못된 HTML 코드가 강조 표시되어 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowImageMaps()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [이미지 맵] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. 이미지 맵이 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowInvisibleElements()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [보이지 않는 요소] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. 보이지 않는 요소 표시자가 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowLayerBorders()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [레이어 테두리] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. 레이어 테두리가 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowLineNumbers()

지원 버전

Dreamweaver 4

설명

이 함수는 [코드] 뷰에 행 번호가 표시되는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 행 번호가 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowRulers()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [눈금자] > [보기] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. 눈금자가 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowSyntaxColoring()

지원 버전

Dreamweaver 4

설명

이 함수는 [문서] 윈도우의 [코드] 뷰에 구문 색상 표시 기능이 켜져 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 구문 색상 표시 기능이 켜져 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowTableBorders()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [표 테두리] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. 표 테두리가 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowToolbar()

지원 버전

Dreamweaver 4

설명

이 함수는 툴바가 표시되는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 툴바가 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowTracingImage()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [바탕 이미지] > [보기] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. 옵션이 켜져 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getShowWordWrap()

지원 버전

Dreamweaver 4

설명

이 함수는 문서 윈도우의 [코드] 뷰에 줄 바꿈 기능이 켜져 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 줄 바꿈 기능이 켜져 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getSnapToGrid()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [격자] > [격자에 스냅] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. [격자에 스냅] 옵션이 켜져 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.setEditNoFramesContent()

지원 버전

Dreamweaver 3

설명

이 함수는 [수정] > [프레임세트] > [비프레임 내용 편집] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수

bEditNoFrames

- **bDuplicate** 인수는 부울 값입니다. **true**로 설정하면 [비프레임 내용 편집] 옵션이 켜지고 **false**로 설정하면 꺼집니다.

반환값

없음

활성자

435페이지의 “[dom.canEditNoFramesContent\(\)](#)”를 참조하십시오.

dom.setHideAllVisualAids()

지원 버전

Dreamweaver 4

설명

이 함수는 [보기] 메뉴의 개별 설정에 관계없이 모든 테두리, 이미지 맵 및 보이지 않는 요소를 숨깁니다.

인수

bSet

- **bSet** 인수는 부울 값입니다. **true**로 설정하면 시각 도구가 숨겨지고 **false**로 설정하면 표시됩니다.

반환값

없음

dom.setPreventLayerOverlaps()

지원 버전

Dreamweaver 3

설명

이 함수는 [레이어 겹침 방지] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수

bPreventLayerOverlaps

- **bPreventLayerOverlaps** 인수는 부울 값입니다. **true**로 설정하면 [레이어 겹침 방지] 옵션이 켜지고 **false**로 설정하면 꺼집니다.

반환값
없음

dom.setShowFrameBorders()

지원 버전
Dreamweaver 3

설명
이 함수는 [보기] > [프레임 테두리] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수
bShowFrameBorders

- **bShowFrameBorders** 인수는 부울 값입니다. true로 설정하면 [프레임 테두리] 옵션이 켜지고 false로 설정하면 꺼집니다.

반환값
없음

dom.setShowGrid()

지원 버전
Dreamweaver 3

설명
이 함수는 [보기] > [격자] > [격자 표시] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수
bShowGrid

- **bShowGrid** 인수는 부울 값입니다. true로 설정하면 [보기] > [격자] > [격자 표시] 옵션이 켜지고 false로 설정하면 꺼집니다.

반환값
없음

dom.setShowHeaderView()

지원 버전
Dreamweaver 3

설명
이 함수는 [보기] > [헤드 내용] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수
bShowHead

- **bShowHead** 인수는 부울 값입니다. true로 설정하면 [헤드 내용] 옵션이 켜지고 false로 설정하면 꺼집니다.

반환값
없음

dom.setShowInvalidHTML()

지원 버전
Dreamweaver 4

설명
이 함수는 문서 윈도우의 [코드] 뷰에서 잘못된 HTML 코드의 강조 표시 기능을 설정하거나 해제합니다.
이 함수는 잘못된 HTML 코드가 강조 표시되어 있는지 여부를 확인합니다.

인수
bShow

- **bShow** 인수는 부울 값입니다. true로 설정하면 잘못된 HTML 코드가 강조 표시되고 false로 설정하면 강조 표시되지 않습니다.

반환값
없음

dom.setShowImageMaps()

지원 버전
Dreamweaver 3

설명
이 함수는 [보기] > [이미지 맵] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수
bShowImageMaps

- **bShowImageMaps** 인수는 부울 값입니다. true로 설정하면 [이미지 맵] 옵션이 켜지고 false로 설정하면 꺼집니다.

반환값
없음

dom.setShowInvisibleElements()

지원 버전
Dreamweaver 3

설명
이 함수는 [보기] > [보이지 않는 요소] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수**bViewInvisibleElements**

- **bViewInvisibleElements** 인수는 부울 값입니다. **true**로 설정하면 [보이지 않는 요소] 옵션이 켜지고 **false**로 설정하면 꺼집니다.

반환값

없음

dom.setShowLayerBorders()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [레이어 테두리] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수**bShowLayerBorders**

- **bShowLayerBorders** 인수는 부울 값입니다. **true**로 설정하면 [레이어 테두리] 옵션이 켜지고 **false**로 설정하면 꺼집니다.

반환값

없음

dom.setShowLineNumbers()

지원 버전

Dreamweaver 4

설명

이 함수는 문서 윈도우의 [코드] 뷰에 행 번호를 표시하거나 숨깁니다.

인수**bShow**

- **bShow** 인수는 부울 값입니다. **true**로 설정하면 행 번호가 표시되고 **false**로 설정하면 숨겨집니다.

반환값

없음

dom.setShowRulers()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [눈금자] > [보기] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수

bShowRulers

- **bShowRulers** 인수는 부울 값입니다. true로 설정하면 [보기] 옵션이 켜지고 false로 설정하면 꺼집니다.

반환값

없음

dom.setShowSyntaxColoring()

지원 버전

Dreamweaver 4

설명

이 함수는 문서 윈도우의 [코드] 뷰에서 구문 색상 표시 기능을 설정하거나 해제합니다.

인수

bShow

- **bShow** 인수는 부울 값입니다. true로 설정하면 구문 색상이 표시되고 false로 설정하면 표시되지 않습니다.

반환값

없음

dom.setShowTableBorders()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [표 테두리] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수

bShowTableBorders

- **bShowTableBorders** 인수는 부울 값입니다. true로 설정하면 [표 테두리] 옵션이 켜지고 false로 설정하면 꺼집니다.

반환값

없음

dom.setShowToolbar()

지원 버전

Dreamweaver 4

설명

이 함수는 툴바를 표시하거나 숨깁니다.

인수

bShow

- **bShow** 인수는 부울 값입니다. **true**로 설정하면 툴바가 표시되고 **false**로 설정하면 표시되지 않습니다.

반환값

없음

dom.setShowTracingImage()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [바탕 이미지] > [보기] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수

bShowTracingImage

- **bShowTracingImage** 인수는 부울 값입니다. **true**로 설정하면 [보기] 옵션이 켜지고 **false**로 설정하면 꺼집니다.

반환값

없음

dom.setShowWordWrap()

지원 버전

Dreamweaver 4

설명

이 함수는 문서 윈도우의 [코드] 뷰에서 줄 바꿈 옵션의 끄기 또는 켜기 설정을 전환합니다.

인수

bShow

- **bShow** 인수는 부울 값입니다. **true**로 설정하면 줄이 바뀌고 **false**로 설정하면 줄이 바뀌지 않습니다.

반환값

없음

dom.setSnapToGrid()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [격자] > [격자에 스냅] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수**bSnapToGrid**

- **bSnapToGrid** 인수는 부울 값입니다. **true**로 설정하면 [격자에 스냅] 옵션이 켜지고 **false**로 설정하면 꺼집니다.

반환값

없음

dreamweaver.getHideAllFloaters()

지원 버전

Dreamweaver 3, CS4에서 업데이트됨

설명

결합된 패널이나 부동 패널이 모두 숨겨지는지 여부를 확인합니다. 결과에 [삽입] 막대의 표시 상태는 포함되지 않습니다. 다음 구성 요소는 숨겨진 것으로 간주되지 않습니다.

- 닫힌 패널
- 축소된 탭 그룹
- 축소된 패널

참고: 결과에 [삽입] 막대는 포함되지 않지만 [삽입] 패널은 포함됩니다.

인수

없음

반환값

부울 값을 반환합니다. 모든 패널이 숨겨지면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.getShowStatusBar()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [상태 표시줄] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. 상태 표시줄이 표시되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.htmlInspector.getShowAutoIndent()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 관리자에서 [자동 들여쓰기] 옵션이 켜져 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 자동 들여쓰기 기능이 켜져 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.htmlInspector.getShowInvalidHTML()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 관리자에서 잘못된 HTML 코드가 강조 표시되어 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 잘못된 HTML 코드가 강조 표시되어 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.htmlInspector.getShowLineNumbers()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 관리자에서 행 번호가 나타나는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 행 번호가 나타나면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.htmlInspector.getShowSyntaxColoring()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 관리자에서 구문 색상 표시 기능이 켜져 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 구문 색상 표시 기능이 켜져 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.htmlInspector.getShowWordWrap()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 관리자에서 줄 바꿈 기능이 켜져 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 줄 바꿈 기능이 켜져 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.htmlInspector.setShowAutoIndent()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 관리자에서 [자동 들여쓰기] 옵션을 켜거나 끕니다.

인수

bShow

- **bShow** 인수는 부울 값입니다. **true**로 설정하면 자동 들여쓰기 기능이 켜지고 **false**로 설정하면 꺼집니다.

반환값

없음

dreamweaver.htmlInspector.setShowInvalidHTML()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 관리자에서 잘못된 HTML 코드의 강조 표시 기능을 켜거나 끕니다.

인수

bShow

- **bShow** 인수는 부울 값입니다. **true**로 설정하면 잘못된 HTML 코드가 강조 표시되고 **false**로 설정하면 강조 표시되지 않습니다.

반환값

없음

dreamweaver.htmlInspector.setShowLineNumbers()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 관리자의 [코드] 뷰에서 행 번호를 표시하거나 숨깁니다.

인수

bShow

- **bShow** 인수는 부울 값입니다. **true**로 설정하면 행 번호가 표시되고 **false**로 설정하면 숨겨집니다.

반환값

없음

dreamweaver.htmlInspector.setShowSyntaxColoring()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 관리자의 [코드] 뷰에서 구문 색상 표시 기능을 켜거나 끕니다.

인수

bShow

- **bShow** 인수는 부울 값입니다. **true**로 설정하면 구문 색상이 표시되고 **false**로 설정하면 표시되지 않습니다.

반환값

없음

dreamweaver.htmlInspector.setShowWordWrap()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 관리자에서 [줄 바꿈] 옵션을 켜거나 끕니다.

인수

bShow

- **bShow** 인수는 부울 값입니다. **true**로 설정하면 줄 바꿈 기능이 켜지고 **false**로 설정하면 꺼집니다.

반환값

없음

dreamweaver.setHideAllFloaters()

지원 버전

Dreamweaver 3, CS4에서 업데이트됨

설명

이 함수는 모든 패널을 표시하거나 숨깁니다. 이 작업은 [삽입] 막대에 영향을 미치지 않습니다.

인수

bShowFloatingPalettes

- **bShowFloatingPalettes** 인수는 부울 값입니다. **true**로 설정하면 모든 패널이 숨겨지고 **false**로 설정하면 표시됩니다. 일부 패널이 표시되는 경우 **false**를 전달하면 나머지 패널이 표시됩니다. 모든 패널이 표시되는 경우 **false**를 전달하면 아무 효과가 없습니다.

참고: 이 명령은 모든 패널이 표시되지 않는 경우 패널을 숨기지 않습니다. 따라서 일부 패널이 표시되는 경우 **true**를 전달하면 아무 효과가 없습니다.

반환값

없음

dreamweaver.setShowStatusBar()

지원 버전

Dreamweaver 3

설명

이 함수는 [보기] > [상태 표시줄] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수

bShowStatusBar

- **bShowStatusBar** 인수는 부울 값입니다. **true**로 설정하면 [상태 표시줄] 옵션이 켜지고 **false**로 설정하면 꺼집니다.

반환값

없음

site.getShowToolTips()

지원 버전

Dreamweaver 3

설명

이 함수는 [도구 설명] 옵션의 현재 상태를 가져옵니다.

인수

없음

반환값

부울 값을 반환합니다. [사이트] 패널에 도구 설명이 표시되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.setShowToolTips()

지원 버전

Dreamweaver 3

설명

이 함수는 [도구 설명] 옵션의 켜기 또는 끄기 설정을 전환합니다.

인수

bShowToolTips

- **bShowToolTips** 인수는 부울 값입니다. **true**로 설정하면 [도구 설명] 옵션이 켜지고 **false**로 설정하면 꺼집니다.

반환값

없음

툴바 함수

다음 JavaScript 함수를 사용하면 툴바 및 툴바 레이블의 표시 여부를 가져오고 설정할 수 있으며, 현재 윈도우에 있는 툴바 항목의 레이블을 얻을 수 있고, 툴바의 위치를 정하거나 툴바 ID를 얻을 수 있습니다. 툴바 만들기 또는 수정에 대한 자세한 내용은 Dreamweaver 확장 도움말에서 "툴바"를 참조하십시오.

dom.getShowToolbarIconLabels()

지원 버전

Dreamweaver MX

설명

이 함수는 버튼 레이블이 현재 문서 윈도우에 표시되는지 여부를 확인합니다. 레이블이 정의된 경우 버튼 컨트롤이 아닌 컨트롤의 레이블은 항상 표시됩니다.

인수

없음

반환값

부울 값을 반환합니다. 버튼의 레이블이 현재 문서 윈도우에 표시되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

다음 예제에서는 버튼의 레이블이 표시되도록 합니다.

```
var dom = dw.getDocumentDom();  
if (dom.getShowToolbarIconLabels() == false)  
{  
    dom.setShowToolbarIconLabels(true);  
}
```

dom.getToolbarIdArray()

지원 버전

Dreamweaver MX

설명

이 함수는 응용 프로그램의 모든 툴바에 대한 ID 배열을 반환합니다. **dom.getToolbarIdArray()**를 사용하여 모든 툴바를 끌 수 있으므로 위치를 다시 지정하고 특정 세트만 표시할 수 있습니다.

인수

없음

반환값

모든 툴바 ID의 배열

예제

다음 예제에서는 **tb_ids** 변수에 툴바 ID의 배열을 저장합니다.

```
var tb_ids = new Array();  
tb_ids = dom.getToolbarIdArray();
```

dom.getToolbarItemValue()

지원 버전

Dreamweaver MX 2004

설명

지정한 툴바 항목의 값을 가져옵니다.

인수

toolbarID, itemID

- **toolbarID** 인수는 값을 알고자 하는 항목이 들어 있는 툴바의 ID를 지정하는 문자열입니다.
- **itemID** 인수는 값을 알고자 하는 항목의 ID를 지정하는 문자열입니다.

반환값

툴바 항목의 값을 나타내는 문자열을 반환합니다.

예제

다음의 `receiveArguments()` 예제는 [크기] 텍스트 필드의 비헤이비어를 제어하는 툴바 명령 안에 있습니다. 이 함수는 [크기] 필드의 값을 인수로 가져온 다음 [단위] 필드의 값을 읽어 CSS 속성 `font-size` 함수에 대한 유효한 값을 생성합니다.

```
receiveArguments(newSize) {
var dom = dw.getDocumentDOM();
if (newSize != ""){
    dom.applyFontMarkupAsStyle('font-size', newSize +
        dom.getToolbarItemValue("DW_Toolbar_Text", "DW_Text_Units"));
}
else{
    dom.removeFontMarkupAsStyle('font-size');
}
}
```

dom.getToolbarLabel()

지원 버전

Dreamweaver MX

설명

이 함수는 지정한 툴바의 레이블을 얻습니다. 툴바를 표시하거나 숨기는 메뉴에 대해 `dom.getToolbarLabel()`을 사용할 수 있습니다.

인수

toolbar_id

- **toolbar_id** 인수는 툴바의 ID이며, `toolbars.xml` 파일에 있는 `toolbar` 태그의 ID 속성 값입니다.

반환값

`toolbar` 태그에 속성으로 할당된 이름 문자열인 **label**을 반환합니다.

예제

다음 예제에서는 `label` 변수에 `myEditbar`의 레이블을 저장합니다.

```
var label = dom.getToolbarLabel("myEditbar");
```

dom.getToolbarVisibility()

지원 버전

Dreamweaver MX

설명

이 함수는 **toolbar_id**로 지정 한 툴바가 표시되는지 여부를 나타내는 부울 값을 반환합니다.

인수

toolbar_id

- **toolbar_id** 인수는 툴바에 할당된 ID 문자열입니다.

반환값

부울 값을 반환합니다. 툴바가 표시되면 **true**를 반환하고 툴바가 표시되지 않거나 없으면 **false**를 반환합니다.

예제

다음 예제에서는 문서 윈도우에 myEditbar라는 툴바가 표시되는지 확인한 다음 해당 값을 retVal 변수에 저장합니다.

```
var retVal = dom.getToolbarVisibility("myEditbar");
return retVal;
```

dom.setToolbarItemAttribute()

지원 버전

Dreamweaver MX 2004

설명

툴바 항목의 세 이미지 속성 또는 tooltip 속성에 대한 속성 값을 변경합니다.

인수

toolbarID, toolbarItemId, attrName, attrValue

- **toolbarID** 인수는 툴바의 ID를 지정하는 문자열입니다.
- **toolbarItemId** 인수는 툴바 항목의 ID를 지정하는 문자열입니다.
- **attrName** 인수는 설정할 속성의 이름을 지정하는 문자열입니다. 유효한 값은 'image', 'overImage', 'disabledImage' 또는 'tooltip'입니다.
- **attrValue** 인수는 설정할 값을 지정하는 문자열입니다.

반환값

없음

예제

다음 예제에서는 dom.setToolbarItemAttribute()를 세 번 호출하여 ID가 DW_Toolbar_Main인 툴바의 툴바 항목 MyButton에 image, imageOver 및 tooltip 속성을 설정합니다.

```
var dom = dw.getDocumentDOM();
dom.setToolbarItemAttribute('DW_Toolbar_Main', 'MyButton', 'image',
    'Toolbars/imgs/newimage.gif');
dom.setToolbarItemAttribute('DW_Toolbar_Main', 'MyButton', 'imageOver',
    'Toolbars/imgs/newimageOver.gif');
dom.setToolbarItemAttribute('DW_Toolbar_Main', 'MyButton', 'tooltip', 'One fine button');
```

dom.setShowToolbarIconLabels()

지원 버전

Dreamweaver MX

설명

이 함수는 레이블이 있는 버튼의 레이블을 표시하도록 설정합니다. 레이블이 정의된 경우 버튼 컨트롤이 아닌 컨트롤의 레이블은 항상 표시됩니다.

인수

bShow

- **bShow** 인수는 부울 값입니다. **true**로 설정하면 버튼의 레이블이 표시되고 **false**로 설정하면 표시되지 않습니다.

반환값

없음

예제

다음 예제에서는 툴바에 있는 버튼의 레이블을 표시하도록 설정합니다.

```
dom.setShowToolbarIconLabels(true);
```

dom.setToolbarPosition()

지원 버전

Dreamweaver MX

설명

이 함수는 지정한 툴바를 지정한 위치로 이동합니다.

참고: 툴바의 현재 위치를 확인할 방법은 없습니다.

인수

toolbar_id, position, relative_to

- **toolbar_id** 인수는 툴바의 ID이며, toolbars.xml 파일에 있는 toolbar 태그의 ID 속성 값입니다.
- **position** 인수는 툴바 위치를 다른 툴바에 상대적으로 지정합니다. **position**에 사용할 수 있는 값은 아래 목록과 같습니다.
 - **top**은 기본 위치입니다. 툴바가 문서 윈도우의 맨 위에 나타납니다.
 - **below**를 지정하면 툴바는 **relative_to**에 지정된 툴바 바로 아래 행의 시작 부분에 표시됩니다. 툴바가 **relative_to**에 지정된 툴바를 찾지 못하면 오류가 보고됩니다.
 - **floating**을 지정하면 툴바가 문서 위에 표시되는 부동 상태가 됩니다. 이때 다른 부동 툴바와 겹치지 않도록 자동으로 툴바 위치가 조절됩니다. Macintosh에서 floating은 top과 같은 방식으로 처리됩니다.

- **relative_to="toolbar_id"**는 **position**이 below로 지정된 경우 필수입니다. 그 외의 경우에는 무시됩니다. 이 툴바가 배치될 지점 위에 있는 툴바의 ID를 지정합니다.

반환값

없음

예제

다음 예제에서는 myEditbar의 위치를 myPicturebar 툴바의 아래로 설정합니다.

```
dom.setToolbarPosition("myEditbar", "below", "myPicturebar");
```

dom.setToolbarVisibility()

지원 버전

Dreamweaver MX

설명

이 함수는 지정한 툴바를 표시하거나 숨깁니다.

인수

toolbar_id, bShow

- **toolbar_id** 인수는 툴바의 ID이며, toolbars.xml 파일에 있는 toolbar 태그의 ID 속성 값입니다.
- **bShow** 인수는 툴바의 표시 여부를 나타내는 부울 값입니다. **bshow**가 true인 경우 dom.setToolbarVisibility()는 툴바를 표시하고, **bShow**가 false인 경우 dom.setToolbarVisibility()는 툴바를 숨깁니다.

반환값

없음

예제

다음 예제에서는 툴바 myEditbar가 문서 윈도우에 표시되는지 확인하여 표시되지 않으면 myEditbar가 표시되도록 설정합니다.

```
var dom = dw.getDocumentDOM();  
if(dom != null && dom.getToolbarVisibility("myEditbar") == false)  
{  
    dom.setToolbarVisibility("myEditbar", true);  
}
```

dreamweaver.reloadToolbars()

지원 버전

Dreamweaver CS4

설명

이 함수는 Configuration/Toolbars 폴더에 있는 모든 JavaScript 툴바를 새로 고칩니다.

인수**{resetToDefault}**

- resetToDefault 인수는 toolbars.xml 파일의 각 툴바에 대한 표시 여부 및 위치 기본값을 읽을지 여부를 나타내는 부울 값입니다. 이 값이 false이거나 제공되지 않으면 툴바 위치 및 표시 여부가 새로 고칠 때 유지됩니다. 이 인수는 선택 사항입니다.

반환값

없음

윈도우 함수

윈도우 함수는 문서 윈도우 및 부동 패널과 관련된 작업을 처리합니다. 윈도우 함수는 부동 패널을 표시하거나 숨기고 문서 윈도우에서 포커스가 있는 부분을 확인하며 활성 문서를 설정합니다. [사이트] 패널과 관련된 작업에 대해서는 198페이지의 “[사이트 함수](#)”를 참조하십시오.

참고: 이 단원에 설명된 함수 중 일부는 Windows에서만 작동합니다. Windows에서만 작동하는지 여부는 함수 설명에 명시됩니다.

dom.getFocus()

지원 버전

Dreamweaver 3

설명

이 함수는 문서에서 현재 포커스가 있는 부분을 확인합니다.

인수

없음

반환값

다음 문자열 중 하나를 반환합니다.

- HEAD 영역이 활성 상태인 경우 "head" 문자열
- BODY 또는 NOFRAMES 영역이 활성 상태 경우 "body" 문자열
- 프레임세트 또는 프레임이 선택된 경우 "frameset" 문자열
- 문서에 포커스가 없는 경우(예: 포커스가 속성 관리자나 다른 부동 패널에 있는 경우) "none" 문자열

dom.getView()

지원 버전

Dreamweaver 4, CS4에서 업데이트됨

설명

이 함수는 표시되어 있는 뷰를 확인합니다.

인수

없음

반환값

현재 표시된 뷰에 따라 design, code, split 또는 "split code"를 반환합니다.

dom.getWindowTitle()

지원 버전

Dreamweaver 3

설명

이 함수는 문서가 포함된 윈도우의 제목을 가져옵니다.

인수

없음

반환값

문서에서 TITLE 태그 사이에 나타나는 텍스트를 포함하는 문자열을 반환하거나, 열려 있는 윈도우에 문서가 없는 경우에는 아무 것도 반환하지 않습니다.

dom.setView()

지원 버전

Dreamweaver 4, CS4에서 업데이트됨

설명

이 함수는 [디자인] 또는 [코드] 뷰를 표시하거나 숨겨서 디자인 전용, 코드 전용 또는 분할 뷰를 표시합니다.

인수

viewString

- *viewString* 인수는 생성할 뷰이며 design, code, split 또는 "split code" 값 중 하나여야 합니다.

반환값

없음

dreamweaver.bringAttentionToFloater()

지원 버전

Dreamweaver MX

설명

지정한 패널 또는 관리자틀 맨 앞으로 가져오고 패널 또는 관리자틀 잘 보이도록 표시하여 주의를 끕니다. 이 함수의 기능은 dw.toggleFloater()와 약간 다릅니다.

인수**floaterName**

- **floaterName** 인수는 윈도우, 패널 또는 관리자의 이름입니다.

반환값

없음

예제

다음 예제에서는 [에셋] 패널을 열고 잘 보이도록 표시합니다.

```
dw.bringAttentionToFloater("library");
```

dreamweaver.cascade()

지원 버전

Dreamweaver MX(Windows에만 해당), Dreamweaver 8(Macintosh 지원이 추가됨)

설명

각 문서 윈도우를 왼쪽 상단 모서리부터 시작하여 아래쪽으로 차례 차례 계단식으로 배열합니다.

인수

없음

반환값

없음

예제

다음 예제에서는 열려 있는 문서를 계단식으로 배치합니다.

```
dw.cascade()
```

dreamweaver.getActiveWindow()

지원 버전

Dreamweaver 3

설명

이 함수는 활성 윈도우의 문서를 가져옵니다.

인수

없음

반환값

활성 윈도우의 문서에 해당하는 문서 객체를 반환하거나, 문서가 프레임에 있는 경우에는 프레임세트에 해당하는 문서 객체를 반환합니다.

dreamweaver.getDocumentList()

지원 버전

Dreamweaver 3

설명

이 함수는 열려 있는 모든 문서의 목록을 가져옵니다.

인수

없음

반환값

열려 있는 각 문서 윈도우에 해당하는 문서 객체의 배열을 반환합니다. 문서 윈도우에 프레임셋이 있는 경우 문서 객체는 프레임의 내용이 아니라 프레임셋을 참조합니다.

dreamweaver.getFloatersVisible()

지원 버전

설명

결합된 패널이나 부동 패널 중 표시되는 것이 있는지 여부를 확인합니다. 다음 항목은 결과에 표시되지 않습니다.

- [삽입] 막대의 표시 상태
- 닫힌 패널

참고: 결과에 [삽입] 막대는 포함되지 않지만 [삽입] 패널은 포함됩니다.

인수

없음

반환값

부울 값을 반환합니다. 패널이 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.getFloaterVisibility()

지원 버전

Dreamweaver 3, CS4에서 업데이트됨

설명

이 함수는 지정된 패널 또는 관리자의 표시 여부를 확인합니다.

인수

floatersName

- floatersName 인수는 부동 패널의 이름입니다. floatersName과 일치하는 내장 패널 이름이 없는 경우 Dreamweaver에서는 Configuration/Floaters 폴더에서 floatersName.htm이라는 파일을 검색합니다. 여기서 floatersName은 부동 패널의 이름입니다.

내장 Dreamweaver 패널의 floatersName 값은 다음 목록에서 패널 이름의 오른쪽에 있는 문자열입니다.

에셋 = "assets"
비헤이비어 = "behaviors"
바인딩 = "data bindings"
코드 관리자 = "html"
구성 요소 = "server components"
CSS 스타일 = "css styles"
데이터베이스 = "databases"
프레임 = "frames"
작업 내역 = "history"
삽입 막대 = "objects"
레이어 = "layers"
라이브러리 = "library"
링크 확인기 결과 = "linkchecker"
속성 = "properties"
참조 = "reference"
보고서 결과 = "reports"
검색 결과 = "search"
선택 항목 관리자 = "selection inspector"
서버 비헤이비어 = "server behaviors"
사이트 = "site"
사이트 파일 = "site files"
코드 단편 = "snippets"
브라우저 호환성 확인 = "bcc"
유효성 검사 결과 = "validation"

반환값

부울 값을 반환합니다. 부동 패널이 표시되어 있고 맨 앞에 나와 있으면 true를 반환하고 그렇지 않거나 floatersName이라는 부동 패널을 찾을 수 없으면 false를 반환합니다.

dreamweaver.getFocus()

지원 버전

Dreamweaver 4

설명

이 함수는 응용 프로그램에서 현재 포커스가 있는 부분을 확인합니다.

인수

bAllowFloaters

- bAllowFloaters 인수는 부울 값입니다. true로 설정하면 부동 패널에 포커스가 있는 경우 함수에서 부동 패널의 이름을 반환하고 false로 설정하면 이를 반환하지 않습니다.

반환값

다음 문자열 중 하나를 반환합니다.

- 문서 윈도우에 포커스가 있는 경우 "document" 문자열
- [사이트] 패널에 포커스가 있는 경우 "site" 문자열
- [텍스트] 뷰에 포커스가 있는 경우 "textView" 문자열
- 코드 관리자에 포커스가 있는 경우 "html" 문자열
- bAllowFloaters가 true이고 부동 패널에 포커스가 있는 경우 floaterName 문자열. 여기서 floaterName은 "objects", "properties", "launcher", "library", "css styles", "html styles", "behaviors", "timelines", "layers", "frames", "templates" 또는 "history"입니다.
- (Macintosh) [사이트] 패널과 문서 윈도우가 모두 열려 있지 않은 경우 "none" 문자열

dreamweaver.getPrimaryView()

지원 버전

Dreamweaver 4

설명

이 함수는 맨 위의 기본 보기로 표시되어 있는 뷰를 확인합니다.

인수

없음

반환값

표시되어 있거나 분할 뷰의 맨 위에 있는 뷰가 무엇인지에 따라 "design" 또는 "code" 문자열을 반환합니다.

dreamweaver.getSnapDistance()

지원 버전

Dreamweaver 4

설명

이 함수는 스냅 거리를 픽셀 단위로 반환합니다.

인수
없음

반환값
스냅 거리를 픽셀 단위로 나타내는 정수를 반환합니다. 기본값은 10픽셀입니다. 0은 스냅 기능이 꺼져 있음을 나타냅니다.

dreamweaver.minimizeRestoreAll()

지원 버전
Dreamweaver 4

설명
이 함수는 Dreamweaver에서 모든 윈도우를 최소화하여 아이콘으로 축소하거나 복원합니다.

인수
bMinimize

- **bMinimize** 인수는 부울 값입니다. true로 설정하면 윈도우가 최소화되고 false로 설정하면 최소화된 윈도우가 복원됩니다.

반환값
없음

dreamweaver.setActiveWindow()

지원 버전
Dreamweaver 3

설명
이 함수는 지정된 문서가 포함된 윈도우를 활성화합니다.

인수
documentObject, {bActivateFrame}

- **documentObject** 인수는 문서의 DOM 트리 루트에 있는 객체(dreamweaver.getDocumentDOM() 함수가 반환하는 값)입니다.
- **bActivateFrame** 인수는 선택적이며 **documentObject**가 프레임세트 안에 있는 경우에만 적용할 수 있습니다.
bActivateFrame 인수는 부울 값입니다. true로 설정하면 문서가 들어 있는 프레임과 프레임세트가 들어 있는 윈도우가 활성화되고 false로 설정하면 활성화되지 않습니다.

반환값
없음

dreamweaver.setFloaterVisibility()

지원 버전
Dreamweaver 3, CS4에서 업데이트됨

설명

이 함수는 특정 부동 패널 또는 관리자의 표시 여부를 지정합니다.

인수

floatName, bIsVisible

- floatName 인수는 부동 패널의 이름입니다. floatName이 내장 패널 이름 중 하나와 일치하지 않으면 Dreamweaver에서는 Configuration/Floaters 폴더에서 floatName.htm이라는 파일을 검색합니다. Dreamweaver에서 floatName이라는 부동 패널을 찾지 못하면 이 함수는 아무 효과가 없습니다.

내장 Dreamweaver 패널의 floatName 값은 다음 목록에서 패널 이름의 오른쪽에 있는 문자열입니다.

에셋 = "assets"
비헤이비어 = "behaviors"
바인딩 = "data sources"
코드 관리자 = "html"
구성 요소 = "server components"
CSS 스타일 = "css styles"
데이터베이스 = "databases"
프레임 = "frames"
작업 내역 = "history"
HTML 스타일 = "html styles"
삽입 막대 = "objects"
레이어 = "layers"
라이브러리 = "library"
링크 확인기 결과 = "linkchecker"
속성 = "properties"
참조 = "reference"
보고서 결과 = "reports"
검색 결과 = "search"
서버 비헤이비어 = "server behaviors"
사이트 = "site"
사이트 파일 = "site files"
코드 단편 = "snippets"
태그 관리자 = "tag inspector"
브라우저 호환성 확인 = "bcc"
템플릿 = "templates"
유효성 검사 결과 = "validation"

- bIsVisible 인수는 부동 패널을 표시할지 여부를 나타내는 부울 값입니다.

반환값

없음

dreamweaver.setPrimaryView()

지원 버전

Dreamweaver 4

설명

이 함수는 지정한 뷰를 문서 윈도우의 맨 위에 표시합니다.

인수

viewString

- **viewString** 인수는 문서 윈도우의 맨 위에 표시할 뷰이며, "design" 또는 "code" 값 중 하나일 수 있습니다.

반환값

없음

dreamweaver.setSnapDistance()

지원 버전

Dreamweaver 4

설명

이 함수는 스냅 거리를 픽셀 단위로 설정합니다.

인수

snapDistance

- **snapDistance** 인수는 스냅 거리를 픽셀 단위로 나타내는 정수입니다. 기본값은 10픽셀입니다. 스냅 기능을 끄려면 0을 지정합니다.

반환값

없음

dreamweaver.showProperties()

지원 버전

Dreamweaver 3

설명

이 함수는 속성 관리자를 표시하고 포커스를 둡니다.

인수

없음

반환값
없음

dreamweaver.tileHorizontally()

지원 버전
Dreamweaver MX(Windows에만 해당), Dreamweaver 8(Macintosh 지원이 추가됨)

설명
문서 윈도우를 서로 겹치지 않도록 가로 방향으로 나란히 배열합니다. 작업 영역을 수직으로 분할하는 것과 비슷합니다.

인수
없음

반환값
없음

예제
다음 예제에서는 열려 있는 문서를 가로 방향으로 나란히 배치합니다.

```
dw.tileHorizontally()
```

dreamweaver.tileVertically()

지원 버전
Dreamweaver MX(Windows에만 해당), Dreamweaver 8(Macintosh 지원이 추가됨)

설명
문서 윈도우를 서로 겹치지 않도록 세로 방향으로 나란히 배열합니다. 작업 영역을 수평으로 분할하는 것과 비슷합니다.

인수
없음

반환값
없음

예제
다음 예제에서는 열려 있는 문서를 세로 방향으로 나란히 배치합니다.

```
dw.tileVertically()
```

dreamweaver.toggleFloater()

지원 버전
Dreamweaver 3

설명

이 함수는 지정한 패널 또는 관리자를 표시하거나 숨기거나 앞으로 가져옵니다.

참고: 이 함수는 `menus.xml` 파일에서만 유효합니다. 부동 패널을 표시하거나 숨기거나 앞으로 가져오려면 `dw.setFloaterVisibility()`를 사용하십시오.

인수**floaterName**

- **floaterName** 인수는 윈도우의 이름입니다. 부동 패널 이름이 `reference`인 경우 [참조] 패널의 보기/숨김 상태는 [코드] 뷰에서 사용자 선택에 따라 업데이트됩니다. 다른 모든 패널은 항상 선택 영역을 추적하지만 [참조] 패널은 사용자가 추적을 시작한 경우에만 [코드] 뷰의 선택 영역을 추적합니다.

반환값

없음

dreamweaver.updateReference()

지원 버전

Dreamweaver 4

설명

이 함수는 [참조] 부동 패널을 업데이트합니다. [참조] 부동 패널이 표시되어 있지 않은 경우 `dw.updateReference()`는 해당 패널을 표시한 다음 업데이트합니다.

인수

없음

반환값

없음

정보 막대 함수

[정보] 막대는 워크플로를 중단시키지 않고 오류 메시지를 표시하는 데 사용됩니다. 다음 [정보] 막대 함수는 오류 메시지가 포함된 [정보] 막대를 숨기거나 표시하는 데 사용됩니다.

dom.showInfoBar()

지원 버전

Dreamweaver CS4

설명

이 함수는 전달된 메시지가 포함된 [정보] 막대를 표시합니다. [정보] 막대가 이미 표시되어 있으면 해당 메시지가 전달되는 새 메시지로 업데이트됩니다. 메시지가 전달되지 않으면 JS 오류가 발생합니다.

인수

메시지입니다.

반환값

없음

dom.hideInfoBar()

지원 버전

Dreamweaver CS4

설명

이 함수는 [정보] 막대를 숨깁니다.

인수

없음

반환값

없음

관련 파일 함수

관련 파일 함수는 활발하게 사용되는 지원 파일과 관련 파일에 쉽게 액세스할 수 있도록 하여 코드 작성자의 편집 환경을 개선합니다.

dreamweaver.getRelatedFiles()

지원 버전

Dreamweaver CS4

설명

이 함수는 모든 관련 파일의 목록을 가져옵니다. 관련 파일은 하위 문서, 소스 HTML 파일 및 생성된 소스 파일일 수 있습니다.

인수

상위 문서와 생성된 소스 파일의 표시 이름을 지정하는 부울 값입니다.

- 메뉴에서 소스 HTML 및 생성된 소스를 표시하려면 **true** 값을 사용합니다.
- 메뉴에서 관련 파일의 실제 이름을 표시하려면 **false** 값을 사용합니다.

반환값

절대 로컬 URL의 형태로 검색된 모든 관련 파일을 포함하는 문자열의 배열을 반환합니다.

dreamweaver.openRelatedFile()

지원 버전

Dreamweaver CS4

설명

현재 문서의 [코드] 뷰에 선택한 관련 파일을 표시합니다.

인수

파일의 절대 로컬 URL인 문자열입니다.

반환값

없음

dreamweaver.getActiveRelatedFilePath()

지원 버전

Dreamweaver CS4

설명

이 함수는 현재 열려 있는 관련 파일의 전체 경로를 가져옵니다.

인수

없음

반환값

관련 파일의 절대 로컬 URL인 문자열을 반환합니다.

수직 분할 뷰 함수

수직 분할 뷰 함수는 코드 및 디자인이나 코드 및 코드 레이아웃 모드를 나란히 볼 수 있도록 합니다. 또한 사용자는 이러한 함수를 사용하여 분할 뷰 및 분할 코드의 수평 및 수직 방향을 선택하고 전환할 수 있습니다.

dreamweaver.getSplitViewOrientation()

지원 버전

Dreamweaver CS4

설명

이 함수는 현재 분할 뷰 방향을 표시합니다. 뷰가 분할되지 않은 경우에도 방향을 검색할 수 있습니다. 이러한 경우 반환값은 뷰가 분할 뷰 또는 분할 코드로 변경될 때의 방향을 나타냅니다.

인수

없음

반환값

방향을 지정하는 문자열 값을 반환합니다. 현재 방향에 따라 vertical 또는 horizontal 값을 반환합니다.

dreamweaver.setSplitViewOrientation()**지원 버전**

Dreamweaver CS4

설명

이 함수는 현재 분할 뷰 방향을 변경합니다. 뷰가 분할되지 않은 경우에도 방향을 변경할 수 있습니다. 이러한 경우 인수는 다음에 뷰가 분할 뷰 또는 분할 코드로 변경될 때의 방향을 나타냅니다.

인수

방향을 나타내는 문자열 값입니다. 방향을 나타내려면 vertical 또는 horizontal을 각각 사용합니다. 이 인수는 필수입니다.

반환값

부울 값을 반환합니다. 성공하면 true를 반환하고 오류가 발생하면 false를 반환합니다.

dreamweaver.getPrimaryView()**지원 버전**

Dreamweaver CS4

설명

이 함수는 기본 보기의 이름을 가져옵니다. 분할 뷰 또는 분할 코드에서 기본 보기는 분할 뷰 방향에 따라 최상위 윈도우나 왼쪽 윈도우입니다.

인수

없음

반환값

다음 값 중 하나인 기본 보기의 이름을 포함하는 문자열을 반환합니다.

값	설명
code	기본 보기가 [코드] 윈도우입니다.
design	기본 보기가 [디자인] 윈도우입니다.
related file	기본 보기가 관련 파일 윈도우입니다. 이 값은 문서 뷰가 분할 코드이고 관련 파일이 열린 경우에 반환됩니다.

dreamweaver.setPrimaryView()**지원 버전**

Dreamweaver CS4

설명

이 함수는 기본 보기를 변경합니다. 분할 뷰 또는 분할 코드에서 기본 보기는 분할 뷰 방향에 따라 최상위 윈도우나 왼쪽 윈도우입니다.

인수

다음 값 중 하나인 기본 보기의 이름을 포함하는 문자열입니다.

값	설명
code	기본 보기가 [코드] 윈도우입니다.
design	기본 보기가 [디자인] 윈도우입니다.
related file	기본 보기가 관련 파일 윈도우입니다. 이 값은 문서 뷰가 분할 코드이고 관련 파일이 열린 경우에 사용됩니다.

반환값

부울 값을 반환합니다. 성공하면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.isRelatedFileViewOpen()

지원 버전

Dreamweaver CS4

설명

이 함수는 뷰에 관련 파일 뷰가 포함되어 있는지 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 관련 파일 뷰가 열려 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

코드 축소 함수

코드 축소 함수를 사용하면 시각적으로 코드를 축소하고 확장할 수 있습니다. 코드의 임의 선택 영역이나 열기 태그와 닫기 태그 사이의 부분을 축소 또는 확장할 수 있습니다. 코드 축소 함수는 dom과 htmlInspector에 모두 존재하지만, 축소된 범위는 코드 뷰와 코드 관리자에서 모두 동일합니다.

dom.collapseFullTag()

지원 버전

Dreamweaver 8

설명

이 함수는 [코드] 뷰의 선택 영역이 열기 태그와 닫기 태그 한 쌍 안에 모두 포함되는지, 아니면 열기 태그와 닫기 태그 한 쌍을 포함하는지를 확인합니다. 후자의 경우라면 이 함수는 열기 태그의 앞에서 시작하고 닫기 태그의 뒤에서 끝나는 코드 부분을 축소합니다. 전자의 경우라면 이 함수는 작동하지 않습니다.

인수**allowCodeFragmentAdjustment**

- 필수적인 **allowCodeFragmentAdjustment** 인수는 부울 값입니다. 값이 true인 경우 이 인수는 현재 아무 영향도 주지 않거나 false 값과 동일한 영향을 줍니다. false인 경우 Dreamweaver에서는 수정 작업 없이 열기 태그 바로 앞에서 시작하고 닫기 태그 바로 뒤에서 끝나는 코드를 축소합니다.

반환값

없음

예제

다음 예제에서는 [코드] 뷰의 현재 선택 영역 안에서 열기 태그 바로 앞에서 시작하고 닫기 태그 바로 뒤에서 끝나는 코드 부분을 축소합니다.

```
var currentDOM = dw.getDocumentDOM();  
currentDOM.collapseFullTag(false);
```

dom.collapseFullTagInverse()

지원 버전

Dreamweaver 8

설명

이 함수는 [코드] 뷰의 선택 영역이 열기 태그와 닫기 태그 한 쌍 안에 모두 포함되는지, 아니면 열기 태그와 닫기 태그 한 쌍을 포함하는지를 확인합니다. 후자의 경우라면 이 함수는 열기 태그의 앞에 있는 코드와 닫기 태그의 뒤에 있는 코드를 축소합니다. 전자의 경우라면 이 함수는 작동하지 않습니다.

인수**allowAdjustmentOfCodeFragments**

- 필수적인 **allowAdjustmentOfCodeFragments** 인수는 부울 값입니다. 이 인수의 값이 true인 경우 Dreamweaver에서는 열기 태그 앞에 있는 코드와 닫기 태그 뒤에 있는 코드의 테두리를 조정하여 고급 축소 기능을 수행합니다. 이 기능은 현재의 들여쓰기 및 간격을 유지합니다. false인 경우 Dreamweaver에서는 열기 태그 앞에 있는 코드 부분과 닫기 태그 뒤에 있는 코드 부분을 선택 영역에서 지정한 대로 정확하게 축소합니다.

반환값

없음

예제

다음 예제에서는 열기 태그의 앞에 있고 닫기 태그의 뒤에 있는 코드의 테두리를 조정하여 고급 축소 기능을 수행합니다. 이 기능은 현재의 들여쓰기 및 간격을 유지합니다.

```
var currentDOM = dw.getDocumentDOM();  
currentDOM.collapseFullTagInverse(true);
```

dom.collapseSelectedCodeFragment()

지원 버전

Dreamweaver 8

설명

이 함수는 [코드] 뷰에서 선택한 코드를 축소합니다. 선택 영역이 이미 축소되어 있으면 이 함수는 작동하지 않습니다.

인수

allowCodeFragmentAdjustment

- 필수적인 **allowCodeFragmentAdjustment**는 부울 값입니다. 이 인수의 값이 true인 경우 Dreamweaver에서는 현재의 선택 영역의 테두리를 수정하여 고급 축소 기능을 수행합니다. 이 기능은 현재의 들여쓰기 및 간격을 유지합니다. false인 경우 Dreamweaver에서는 현재 선택한 코드 부분을 선택 영역에서 지정하는 대로 정확히 축소합니다.

반환값

없음

예제

다음 예제에서는 [코드] 뷰에서 선택한 코드 부분을 아무런 수정 작업 없이 축소합니다.

```
var currentDOM = dw.getDocumentDOM();  
currentDOM.collapseSelectedCodeFragment(false);
```

dom.collapseSelectedCodeFragmentInverse()

지원 버전

Dreamweaver 8

설명

이 함수는 [코드] 뷰에서 선택한 코드의 앞에 있는 모든 코드와 뒤에 있는 모든 코드를 축소합니다.

인수

allowAdjustmentOfCodeFragments

- 필수적인 **allowAdjustmentOfCodeFragments** 인수는 부울 값입니다. 이 인수의 값이 true인 경우 Dreamweaver에서는 현재 선택 영역의 앞에 있는 코드와 뒤에 있는 코드의 테두리를 조정하여 고급 축소 기능을 수행합니다. 이 기능은 현재의 들여쓰기 및 간격을 유지합니다. false인 경우 Dreamweaver에서는 코드 부분을 선택 영역에서 지정하는 대로 정확히 축소합니다.

반환값

없음

예제

다음 예제에서는 [코드] 뷰에서 조정 작업을 수행한 후 선택한 코드의 앞에 있는 모든 코드와 뒤에 있는 모든 코드를 축소합니다.

```
var currentDOM = dw.getDocumentDOM();  
currentDOM.collapseSelectedCodeFragmentInverse(true);
```

dom.expandAllCodeFragments()

지원 버전

Dreamweaver 8

설명

이 함수는 [코드] 뷰에서 중첩되어 축소된 코드 부분을 포함하여 축소된 모든 코드 부분을 확장합니다.

인수

없음

반환값

없음

예제

다음 예제에서는 [코드] 뷰에서 축소된 모든 코드를 확장합니다.

```
var currentDOM = dw.getDocumentDOM();  
currentDOM.expandAllCodeFragments();
```

dom.expandSelectedCodeFragments()

지원 버전

Dreamweaver 8

설명

이 함수는 [코드] 뷰에서 현재 선택 영역 안에 있는 축소된 모든 코드 부분을 확장합니다. 선택 영역이 이미 확장되어 있으면 이 함수는 작동하지 않습니다.

인수

없음

반환값

없음

예제

다음 예제에서는 [코드] 뷰에서 현재 선택 영역 안에 있는 축소된 모든 코드를 확장합니다.

```
var currentDOM = dw.getDocumentDOM();  
currentDOM.expandSelectedCodeFragments();
```

dreamweaver.htmlInspector.collapseFullTag()

지원 버전

Dreamweaver 8

설명

이 함수는 코드 관리자의 선택 영역이 열기 태그와 닫기 태그 한 쌍 안에 모두 포함되는지, 아니면 열기 태그와 닫기 태그 한 쌍을 포함하는지를 확인합니다. 후자의 경우라면 이 함수는 열기 태그의 앞에서 시작하고 닫기 태그의 뒤에서 끝나는 코드 부분을 축소합니다. 전자의 경우라면 이 함수는 작동하지 않습니다.

인수**allowACodeFragmentAdjustment**

- 필수적인 **allowCodeFragmentAdjustment** 인수는 부울 값입니다. 값이 true인 경우 이 인수는 현재 아무 영향도 주지 않거나 false 값과 동일한 영향을 줍니다. false인 경우 Dreamweaver에서는 수정 작업 없이 열기 태그 바로 앞에서 시작하고 닫기 태그 바로 뒤에서 끝나는 코드를 축소합니다.

반환값

없음

예제

다음 예제에서는 코드 관리자의 현재 선택 영역 안에서 열기 태그 바로 앞에서 시작하고 닫기 태그 바로 뒤에서 끝나는 코드 부분을 축소합니다.

```
dreamweaver.htmlInspector.collapseFullTag(false);
```

dreamweaver.htmlInspector.collapseFullTagInverse()

지원 버전

Dreamweaver 8

설명

이 함수는 코드 관리자의 선택 영역이 열기 태그와 닫기 태그 한 쌍 안에 모두 포함되는지, 아니면 열기 태그와 닫기 태그 한 쌍을 포함하는지를 확인합니다. 후자의 경우라면 이 함수는 열기 태그의 바로 앞에서 시작하고 닫기 태그의 뒤에서 끝나는 코드 부분을 축소합니다. 전자의 경우라면 이 함수는 작동하지 않습니다.

인수**allowAdjustmentOfCodeFragments**

- 필수적인 **allowAdjustmentOfCodeFragments** 인수는 부울 값입니다. 이 인수의 값이 true인 경우 Dreamweaver에서는 열기 태그의 앞에 있는 코드와 닫기 태그의 뒤에 있는 코드의 테두리를 조정하여 고급 축소 기능을 수행합니다. 이 기능은 기존의 들여쓰기 및 간격을 유지합니다. false인 경우 Dreamweaver에서는 수정 작업 없이 열기 태그의 앞에 있는 코드와 닫기 태그의 뒤에 있는 코드를 축소합니다.

반환값

없음

예제

다음 예제에서는 현재 선택 영역의 열기 태그 앞과 닫기 태그 뒤에 있는 코드 섹션에 대해 고급 축소 기능을 수행합니다.

```
dreamweaver.htmlInspector.collapseFullTagInverse(true);
```

dreamweaver.htmlInspector.collapseSelectedCodeFragment()

지원 버전

Dreamweaver 8

설명

이 함수는 코드 관리자에서 선택한 코드를 축소합니다. 선택 영역이 이미 축소되어 있으면 이 함수는 작동하지 않습니다.

인수

allowCodeFragmentAdjustment

- 필수적인 **allowCodeFragmentAdjustment**는 부울 값입니다. 이 인수의 값이 true인 경우 Dreamweaver에서는 현재 선택 영역을 수정하여 고급 축소 기능을 수행합니다. 이 기능은 기존의 들여쓰기 및 간격을 유지합니다. false인 경우 Dreamweaver에서는 현재 선택한 코드 부분을 선택 영역에서 지정하는 대로 정확히 축소합니다.

반환값

없음

예제

다음 예제에서는 코드 관리자에서 조정 작업을 수행한 후 선택한 코드를 축소합니다.

```
dreamweaver.htmlInspector.collapseSelectedCodeFragment (true) ;
```

dreamweaver.htmlInspector.collapseSelectedCodeFragmentInverse())

지원 버전

Dreamweaver 8

설명

이 함수는 코드 관리자에서 선택한 코드의 앞에 있는 모든 코드와 뒤에 있는 모든 코드를 축소합니다. 선택 영역이 이미 축소되어 있으면 이 함수는 작동하지 않습니다.

인수

allowAdjustmentOfCodeFragments

- 필수적인 **allowAdjustmentOfCodeFragments** 인수는 부울 값입니다. 이 인수의 값이 true 경우 Dreamweaver에서는 현재 선택 영역의 앞에 있는 코드 섹션과 뒤에 있는 코드 섹션의 테두리를 조정하여 고급 축소 기능을 수행합니다. 이 기능은 현재의 들여쓰기 및 간격을 유지합니다. false인 경우 Dreamweaver에서는 코드 섹션을 선택 영역에서 지정하는 대로 정확히 축소합니다.

반환값

없음

예제

다음 예제에서는 코드 관리자에서 선택한 코드의 앞에 있는 모든 코드와 뒤에 있는 모든 코드를 선택 영역에서 지정하는 대로 정확히 축소합니다.

```
dreamweaver.htmlInspector.collapseSelectedCodeFragmentInverse (false) ;
```

dreamweaver.htmlInspector.expandAllCodeFragments()

지원 버전

Dreamweaver 8

설명

이 함수는 코드 관리자에서 중첩되어 축소된 코드 부분을 포함하여 축소된 모든 코드 부분을 확장합니다.

인수

없음

반환값

없음

예제

다음 예제에서는 코드 관리자에서 축소된 모든 코드를 확장합니다.

```
dreamweaver.htmlInspector.expandAllCodeFragments();
```

dreamweaver.htmlInspector.expandSelectedCodeFragments()

지원 버전

Dreamweaver 8

설명

이 함수는 코드 관리자에서 현재 선택 영역 안에 있는 축소된 모든 코드 부분을 확장합니다. 선택 영역이 이미 확장되어 있으면 이 함수는 작동하지 않습니다.

인수

없음

반환값

없음

예제

다음 예제에서는 코드 관리자에서 현재 선택 영역 안에 있는 축소된 모든 코드를 확장합니다.

```
dreamweaver.htmlInspector.expandSelectedCodeFragments();
```

코드 뷰 툴바 함수

코드 뷰 툴바 함수를 사용하면 [코드] 뷰에서 텍스트를 삽입하고, 주석을 제거하며, 공백 문자에 대한 특수 문자를 표시하거나 숨길 수 있고, 현재 문서의 경로를 가져올 수 있습니다.

참고: [코딩] 툴바에는 [코드] 뷰에 대한 툴바와 코드 관리자에 대한 툴바의 두 가지가 있습니다. 두 코딩 툴바 모두 Configuration/Toolbars/toolbars.xml 파일에 사용자 정의되어 있습니다.

dom.getOpenPathName()

지원 버전

Dreamweaver 8

설명

이 함수는 열려 있는 문서의 절대 파일 경로를 가져옵니다.

인수

없음

반환값

열려 있는 문서의 절대 파일 경로인 문자열을 반환합니다.

예제

다음 예제에서는 현재 열려 있는 문서의 경로를 포함하는 문자열을 `fileName` 변수에 할당합니다.

```
var fileName = dom.getOpenPathName();
```

dom.getShowHiddenCharacters()

지원 버전

Dreamweaver 8

설명

이 함수는 공백 문자에 대한 특수 문자가 [문서] 윈도우의 [코드] 뷰에 표시되는지 여부를 확인합니다.

인수

없음

반환값

부울 값, 숨긴 문자가 표시되면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

예제

다음 예제에서는 처음에 특수 문자 표시 기능이 켜져 있으면 공백 문자에 대한 특수 문자 표시 기능을 끕니다.

```
var currentDOM = dw.getDocumentDOM();  
if (currentDOM.getShowHiddenCharacters()) {  
    currentDOM.setShowHiddenCharacters(false);  
}
```

dom.setShowHiddenCharacters()

지원 버전

Dreamweaver 8

설명

이 함수는 코드 관리자의 [코드] 뷰에서 공백 문자에 대한 특수 문자를 표시하거나 숨깁니다.

예제는 192페이지의 “[dom.getShowHiddenCharacters\(\)](#)”를 참조하십시오.

인수

show

- 필수적인 **show** 인수는 숨긴 문자를 표시할지 여부를 나타내는 부울 값입니다.

반환값

없음

dom.source.applyComment()

지원 버전

Dreamweaver 8

설명

이 함수는 **beforeText** 인수에 지정된 텍스트를 현재 선택 영역 앞에 삽입하고 **afterText** 인수에 지정된 텍스트를 현재 선택 영역 뒤에 삽입합니다. 그런 다음 이 함수는 현재의 선택 영역을 확장하여 추가된 텍스트를 포함합니다. 그러나 현재 선택 영역이 없으면 이 함수는 아무 것도 선택하지 않습니다. **afterText** 인수에 지정된 텍스트가 null이면 이 함수는 **beforeText** 인수에 지정된 텍스트를 현재 선택 영역에 있는 모든 행의 처음에 삽입합니다.

인수

beforeText, afterText

- beforeText**는 필수 인수입니다. 이 인수는 선택 영역의 처음에 삽입할 텍스트를 지정합니다. **afterText** 인수의 값이 null인 경우에는 선택 영역에 있는 모든 행의 처음에 삽입할 텍스트를 지정합니다.
- 선택적인 **afterText** 인수는 선택 영역의 끝에 삽입할 텍스트를 지정합니다.

반환값

없음

예제

다음 예제에서는 현재 선택 영역을 HTML 주석으로 만듭니다.

```
dw.getDocumentDOM().source.applyComment('<!--', '-->')
```

dom.source.refreshVariableCodeHints()

지원 버전

Dreamweaver CS3

설명

페이지를 다시 검색하여 변수와 해당 클래스 연결을 찾습니다. 이 함수는 색상 상태 엔진과 변수 목록을 다시 구성합니다.

인수

bSyncDoc

- 부울 값입니다. 기본값은 false입니다. true로 설정된 경우 디자인 뷰가 코드 뷰와 동기화됩니다.

반환값

없음

예제

```
dom.source.refreshVariableCodeHints();
```

dom.source.removeComment()

지원 버전

Dreamweaver 8

설명

이 함수는 주석을 제거합니다. 인수를 지정하지 않는 경우 이 함수는 현재 선택 영역에서 서버측 포함 부분과 Dreamweaver 고유의 주석을 제외한 모든 유형의 주석을 제거합니다. 중첩된 주석이 있는 경우 외부 주석만 제거되고, 현재 선택 영역이 없는 경우에는 커서가 위치한 행의 첫 번째 행 주석만 제거됩니다. 인수를 지정하면 이 함수는 **beforeText** 인수와 **afterText** 인수에 지정된 값과 일치하는 주석만 제거하며 해당 주석이 다른 유형의 주석 안에 중첩되어 있는 경우에도 해당 주석만 제거합니다.

인수

beforeText, **afterText**

- **beforeText** 인수는 선택 사항입니다. 이 인수는 선택 영역에서 제거할 주석의 처음 부분을 확인할 텍스트를 지정합니다. **afterText** 인수의 값이 **null**인 경우에는 현재 선택 영역에서 제거할 행 주석의 유형을 지정합니다.
- 선택적인 **afterText** 인수는 선택 영역에서 제거할 주석의 끝을 식별할 텍스트를 지정합니다.

반환값

없음

예제

다음 예제에서는 HTML 주석을 제거합니다.

```
dw.getDocumentDOM().source.removeComment('<!--', '-->')
```

dreamweaver.htmlInspector.getShowHiddenCharacters()

지원 버전

Dreamweaver 8

설명

이 함수는 공백 문자에 대한 특수 문자가 코드 관리자의 [코드] 뷰에 표시되는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 숨긴 문자가 표시되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

다음 예제에서는 처음에 특수 문자 표시 기능이 켜져 있으면 코드 관리자에서 공백 문자에 대한 특수 문자 표시 기능을 끕니다.

```
if (dreamweaver.htmlInspector.getShowHiddenCharacters()) {
    dreamweaver.htmlInspector.setShowHiddenCharacters(false);
}
```

dreamweaver.htmlInspector.setShowHiddenCharacters()**지원 버전**

Dreamweaver 8

설명

이 함수는 코드 관리자의 [코드] 뷰에서 공백 문자에 대한 특수 문자를 표시하거나 숨깁니다.

인수**show**

- 필수적인 **show** 인수는 공백 문자에 대한 숨긴 문자를 표시할지 여부를 나타내는 부울 값입니다.

반환값

부울 값, 숨긴 문자가 표시되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

194페이지의 “[dreamweaver.htmlInspector.getShowHiddenCharacters\(\)](#)”를 참조하십시오.

색상 함수

다음 색상 함수를 사용하여 Extension이 응용 프로그램 사용자 인터페이스와 동일한 스킨을 갖도록 할 수 있습니다.

dreamweaver.getPanelColor()**지원 버전**

Dreamweaver CS4

설명

이 함수는 응용 프로그램 사용자 인터페이스의 패널 색상을 검색합니다. 이러한 색상을 Extension의 패널 색상으로 사용할 수 있습니다. 이 함수는 Extension의 패널 색상이 응용 프로그램 사용자 인터페이스의 패널 색상과 혼합되도록 하는 데 도움이 됩니다.

인수

없음

반환값

다음 값이 포함된 크기가 4인 문자열의 배열을 반환합니다.

- 빨강

- 녹색
- 파랑
- 알파

예제

```
var panelColorArray = dw.getPanelColor();
```

이 예제의 반환값은 다음과 같습니다.

- panelColorArray[0] : 빨강
- panelColorArray[1] : 녹색
- panelColorArray[2] : 파랑
- panelColorArray[3] : 알파

dreamweaver.getAppBarColor()

지원 버전

Dreamweaver CS4

설명

이 함수는 사용자 인터페이스의 응용 프로그램 막대 색상을 검색합니다. 이러한 색상을 **Extension**의 막대 색상으로 사용할 수 있습니다. 이 함수는 **Extension**의 막대 색상이 응용 프로그램 사용자 인터페이스의 막대 색상과 혼합되도록 하는 데 도움이 됩니다.

인수

없음

반환값

다음 값이 포함된 크기가 4인 문자열의 배열을 반환합니다.

- 빨강
- 녹색
- 파랑
- 알파

예제

```
var appBarColorArray = dw.getAppBarColor();
```

이 예제의 반환값은 다음과 같습니다.

- appBarColorArray[0] : 빨강
- appBarColorArray[1] : 녹색
- appBarColorArray[2] : 파랑
- appBarColorArray[3] : 알파

13장: 사이트

Adobe® Dreamweaver® CS4 사이트 함수는 웹 사이트 관리와 관련된 작업을 수행합니다. 이러한 작업에는 보고서 사용자 정의, 새 사이트 정의, 파일 체크 인 및 체크 아웃, 사이트에서의 유효성 검사 실행 등이 있습니다.

보고 함수

보고 함수를 사용하면 보고 기능에 액세스할 수 있으므로 보고 프로세스를 시작하고 감시하며 사용자 정의할 수 있도록 합니다. 자세한 내용은 **Dreamweaver** 확장 도움말에서 "보고서"를 참조하십시오.

dreamweaver.isReporting()

지원 버전

Dreamweaver 4

설명

현재 보고 프로세스가 실행되고 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 프로세스가 실행 중이면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.showReportsDialog()

지원 버전

Dreamweaver 4

설명

[보고서] 대화 상자를 엽니다.

인수

없음

반환값

없음

사이트 함수

사이트 함수는 사이트 파일에서 파일과 관련된 작업을 처리합니다. 사이트 함수를 사용하면 다음 작업을 수행할 수 있습니다.

- 파일 간에 링크 만들기
- 파일 내려받기, 올리기, 체크 인, 체크 아웃
- 파일 선택 및 선택 취소
- 파일 만들기 및 제거
- 사용자가 정의한 사이트 정보 가져오기
- 사이트 정보 가져오기 및 내보내기

dom.getSiteURLPrefixFromDoc()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 정의] 대화 상자의 [로컬 정보] 섹션에 정의된 HTTP 주소에서 추출된 사이트 URL 접두어를 가져옵니다.

인수

없음

반환값

사이트 URL 접두어를 지정하는 문자열을 반환합니다.

예제

다음 예제에서는 현재 문서에 대한 사이트 URL 접두어를 가져옵니다.

```
var currentDOM = dw.getDocumentDOM();  
var sitePrefix = dom.getSiteURLPrefixFromDoc();
```

dom.localPathToSiteRelative()

지원 버전

Dreamweaver 8

설명

이 함수는 로컬 파일 경로를 사이트 상대 URI 참조로 변환합니다.

인수

localFilePath

- 필수적인 **localFilePath** 속성은 로컬 컴퓨터의 로컬 파일에 대한 경로를 포함하는 문자열입니다.

반환값

사이트 상대 URI를 지정하는 문자열을 반환합니다.

예제

다음 예제에서는 [사이트 정의] 대화 상자의 [로컬 정보] 섹션에 지정된 사이트 매핑과 HTTP 주소에 따라 "/myWebApp/myFile.cfm"을 반환합니다.

```
var siteRelativeURI = site.localPathToSiteRelative("C:\Inetpub\wwwroot\siteA\myFile.cfm")
```

dom.siteRelativeToLocalPath()

지원 버전

Dreamweaver 8

설명

이 함수는 사이트 상대 URI 참조를 로컬 파일 경로로 변환합니다.

인수

siteRelativeURI

- 필수적인 **siteRelativeURI** 속성은 사이트 상대 URI를 포함하는 문자열입니다.

반환값

로컬 컴퓨터의 로컬 파일에 대한 경로를 지정하는 문자열을 반환합니다.

예제

다음 예제에서는

```
var filePath = siteRelativeToLocalPath("/myWebApp/myFile.xml");
```

[사이트 정의] 대화 상자의 [로컬 정보] 섹션에 지정된 사이트 매핑과 HTTP 주소에 따라 "C:\Inetpub\wwwroot\siteA\myFile.xml"을 반환합니다.

dreamweaver.compareFiles()

지원 버전

Dreamweaver 8

설명

이 함수는 사용자가 [환경 설정] 대화 상자의 [Diff] 섹션에 설치한 파일 비교 도구를 실행합니다.

인수

file1, file2

- 필수적인 **file1** 속성은 비교할 첫 번째 파일의 전체 경로를 포함하는 문자열입니다.
- 필수적인 **file2** 속성은 비교할 두 번째 파일의 전체 경로를 포함하는 문자열입니다.

반환값

없음

예제

다음 예제에서는 red.htm과 blue.htm을 비교합니다.

```
dw.compareFiles(hc:\data\red.htm", "e:\data\blue.htm");
```

dreamweaver.loadSitesFromPrefs()

지원 버전

Dreamweaver 4

설명

모든 사이트의 사이트 정보를 시스템 레지스트리(Windows)나 Dreamweaver 환경 설정 파일(Macintosh)에서 Dreamweaver로 로드합니다. 이 함수가 호출될 때 사이트가 원격 서버에 연결되어 있으면, 사이트의 연결이 자동으로 해제됩니다.

인수

없음

반환값

없음

dreamweaver.saveSitesToPrefs()

지원 버전

Dreamweaver 4

설명

사용자가 정의한 각 사이트의 정보를 시스템 레지스트리(Windows)나 Dreamweaver 환경 설정 파일(Macintosh)에 모두 저장합니다.

인수

없음

반환값

없음

dreamweaver.siteSyncDialog.compare()

지원 버전

Dreamweaver 8

설명

이 함수는 [환경 설정] 대화 상자의 [파일 비교] 범주에 지정된 파일 비교 응용 프로그램을 실행하여 로컬 및 원격 사이트에서 선택한 파일을 비교합니다.

인수

없음

반환값

없음

활성자

456페이지의 “[dreamweaver.siteSyncDialog.canCompare\(\)](#)”를 참조하십시오.

dreamweaver.siteSyncDialog.markDelete()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 선택된 항목의 액션을 [삭제]로 변경합니다.

인수

없음

반환값

없음

활성자

456페이지의 “[dreamweaver.siteSyncDialog.canMarkDelete\(\)](#)”를 참조하십시오.

dreamweaver.siteSyncDialog.markGet()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 선택된 항목의 액션을 [내려받기]로 변경합니다.

인수

없음

반환값

없음

활성자

457페이지의 “[dreamweaver.siteSyncDialog.canMarkGet\(\)](#)”을 참조하십시오.

dreamweaver.siteSyncDialog.markIgnore()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 선택된 항목의 액션을 [무시]로 변경합니다.

인수

없음

반환값

없음

활성자

457페이지의 “[dreamweaver.siteSyncDialog.canMarkIgnore\(\)](#)”를 참조하십시오.

dreamweaver.siteSyncDialog.markPut()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 선택된 항목의 액션을 [올리기]로 변경합니다.

인수

없음

반환값

없음

활성자

457페이지의 “[dreamweaver.siteSyncDialog.canMarkPut\(\)](#)”을 참조하십시오.

dreamweaver.siteSyncDialog.markSynced()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 선택된 항목의 액션을 [동기화]로 변경합니다.

인수

없음

반환값

없음

활성자

458페이지의 “[dreamweaver.siteSyncDialog.canMarkSynced\(\)](#)”를 참조하십시오.

dreamweaver.siteSyncDialog.toggleShowAllFiles()

지원 버전

Dreamweaver 8

설명

이 함수를 사용하면 [사이트 동기화] 미리 보기 대화 상자에 원격 사이트와 로컬 사이트에서 동일하다고 파악되는 파일이 표시됩니다. [모든 파일 표시] 체크 상자를 선택하고 이 함수를 호출하면 이 체크 상자 선택이 해제되고, 반대로 [모든 파일 표시] 체크 상자를 선택하지 않고 이 함수를 호출하면 이 체크 상자가 선택됩니다.

인수

없음

반환값

없음

site.addLinkToExistingFile()

지원 버전

Dreamweaver 3

설명

[HTML 파일 선택] 대화 상자를 열어 사용자가 파일을 선택하도록 하고, 선택 문서와 해당 파일과의 링크를 생성합니다.

인수

없음

반환값

없음

site.changeLinkSitewide()

지원 버전

Dreamweaver 3

설명

[사이트 전체 링크 바꾸기] 대화 상자를 엽니다.

인수

없음

반환값

없음

site.changeLink()

지원 버전

Dreamweaver 3

설명

[HTML 파일 선택] 대화 상자를 열어 사용자가 링크에 사용할 새 파일을 선택하도록 합니다.

인수

없음

반환값

없음

site.checkIn()

지원 버전

Dreamweaver 3

설명

선택한 파일을 체크 인하고 종속 파일을 다음 중 한 가지 방법으로 처리합니다.

- 사용자가 [사이트 FTP] 환경 설정에서 [올리기/체크 인할 때 확인]을 선택한 경우 [종속 파일] 대화 상자가 나타납니다.
- 사용자가 이전에 [종속 파일] 대화 상자에서 [다시 나타내지 않음] 옵션을 선택하고 [예]를 클릭한 경우에는 대화 상자가 나타나지 않고 종속 파일이 업로드됩니다.
- 사용자가 이전에 [종속 파일] 대화 상자에서 [다시 나타내지 않음] 옵션을 선택하고 [아니오]를 클릭한 경우에는 대화 상자도 나타나지 않고 종속 파일도 업로드되지 않습니다.

인수

siteOrURL

- **siteOrURL** 인수는 "site" 키워드여야 하며, 이는 함수가 단일 파일의 URL 또는 [파일] 패널의 선택 영역에 적용되어야 함을 나타냅니다.

반환값

없음

활성자

459페이지의 “[site.canCheckIn\(\)](#)”을 참조하십시오.

site.checkLinks()

지원 버전

Dreamweaver 3

설명

[링크 확인기] 대화 상자를 열고 지정된 파일의 링크를 확인합니다.

인수**scopeOfCheck**

- **scopeOfCheck** 인수는 링크를 확인할 범위를 지정합니다. 값은 "document", "selection" 또는 "site"여야 합니다.

반환값

없음

site.checkOut()

지원 버전

Dreamweaver 3

설명

선택한 파일을 체크 아웃하고 종속 파일을 다음 중 한 가지 방법으로 처리합니다.

- 사용자가 [사이트 FTP] 환경 설정에서 [내려받기/체크 아웃할 때 확인]을 선택한 경우 [종속 파일] 대화 상자가 나타납니다.
- 사용자가 이전에 [종속 파일] 대화 상자에서 [다시 나타내지 않음] 옵션을 선택하고 [예]를 클릭한 경우에는 대화 상자가 나타나지 않고 종속 파일이 다운로드됩니다.
- 사용자가 이전에 [종속 파일] 대화 상자에서 [다시 나타내지 않음] 옵션을 선택하고 [아니오]를 클릭한 경우에는 대화 상자도 나타나지 않고 종속 파일도 다운로드되지 않습니다.

인수**siteOrURL**

- **siteOrURL** 인수는 "site" 키워드여야 하며, 이는 함수가 단일 파일의 URL 또는 [파일] 패널의 선택 영역에 적용되어야 함을 나타냅니다.

반환값

없음

활성자

459페이지의 "[site.canCheckOut\(\)](#)"을 참조하십시오.

site.checkTargetBrowsers()

지원 버전

Dreamweaver 3

설명

선택한 파일에 대해 대상 브라우저 확인 작업을 수행합니다.

인수

없음

반환값

없음

site.cloak()

지원 버전

Dreamweaver MX

설명

[파일] 패널의 현재 선택 영역이나 지정된 폴더를 숨깁니다.

인수

siteOrURL

siteOrURL 인수는 다음 두 값 중 하나를 포함해야 합니다.

- "site" 키워드 - cloak()이 [파일] 패널의 선택 영역에 적용되어야 함을 나타냅니다..
- 특정 폴더의 URL - cloak()이 지정된 폴더와 해당 폴더의 모든 내용에 적용되어야 함을 나타냅니다.

반환값

없음

활성자

460페이지의 “[site.canCloak\(\)](#)”을 참조하십시오.

site.compareFiles()

지원 버전

Dreamweaver 8

설명

이 함수는 Diff 도구 통합 응용 프로그램을 실행하여 두 파일을 비교합니다.

인수

url

필수적인 **url** 인수는 다음 두 값 중 하나를 포함해야 합니다.

- "site" 키워드 - compare()가 [파일] 패널의 선택 영역에 적용되어야 함을 나타냅니다.
- 원격 버전과 비교할 로컬 파일의 URL

반환값

부울 값을 반환합니다. 비교가 성공하면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

활성자

460페이지의 “[site.canCompareFiles\(\)](#)”를 참조하십시오.

예제

다음 예제에서는 [파일] 패널에서 선택한 파일과 원격 버전을 비교합니다.

```
site.compareFiles("site");
```

site.defineSites()**지원 버전**

Dreamweaver 3

설명

이 함수는 [사이트 정의] 대화 상자를 엽니다.

인수

없음

반환값

없음

site.deleteSelection()**지원 버전**

Dreamweaver 3

설명

선택한 파일을 삭제합니다.

인수

없음

반환값

없음

site.deployFilesToTestingServerBin()**지원 버전**

Dreamweaver MX

설명

지정된 파일을 테스트 서버의 **bin** 폴더에 올립니다. 현재 사이트에 지원 파일 배포에 대한 설정이 정의되어 있지 않은 경우 이 함수는 [테스트 서버에 지원 파일 배포] 대화 상자를 시작합니다.

인수

filesToDeploy

- **filesToDeploy** 인수는 배포될 파일 이름의 배열입니다.

반환값

부울 값을 반환합니다. 파일이 성공적으로 배포되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

이 예제에서는 image1.jpg 및 script1.js 파일을 테스트 서버의 bin 폴더에 배포합니다.

```
site.deployFilesToTestingServerBin("image1.jpg", "script1.js");
```

site.displaySyncInfoForFile()**지원 버전**

Dreamweaver CS3

설명

전달된 매개 변수에 해당하는 파일의 로컬, 원격, 테스트 시간을 포함하는 대화 상자를 표시합니다. 이 정보는 동기화 파일인 dwsync.xml에 저장됩니다.

이 대화 상자에는 다음 네 가지 시간이 표시됩니다.

- [로컬 원격 시간]은 로컬 파일을 원격 서버에 올리거나 내려받는 명령을 마지막으로 실행한 시간 스탬프를 나타냅니다.
- [원격 시간]은 원격 서버의 파일을 원격 서버에 올리거나 내려받는 명령을 마지막으로 실행한 시간 스탬프를 나타냅니다.
- [로컬 테스트 시간]은 로컬 파일을 테스트 서버에 올리거나 내려받는 명령을 마지막으로 실행한 시간 스탬프를 나타냅니다..
- [테스트 시간]은 테스트 서버의 파일을 테스트 서버에 올리거나 내려받는 명령을 마지막으로 실행한 시간 스탬프를 나타냅니다.

dwsync.xml 파일에 해당 파일의 정보가 없는 경우에는 사용 가능한 정보가 없음을 알리는 메시지가 표시됩니다. XML 파일에 시간이 설정된 경우에는 해당 로컬의 날짜/시간 형식으로 표시됩니다(예: 6/24/05 2:43pm). 파일의 해당 항목에 시간이 설정되지 않은 경우에는 대시(-)가 표시됩니다.

이 함수는 'site'가 전달되면 로컬 파일 뷰에 선택된 파일에 적용되고, URL이 전달된 경우에는 로컬 URL에 해당하는 파일에 적용됩니다.

인수

path, 'site'

- path는 로컬 파일에 대한 URL입니다.
- 'site'는 해당 함수가 [파일] 패널에 선택된 파일을 사용함을 나타냅니다.

반환값

없음

활성자

461 페이지의 “[site.canDisplaySyncInfoForFile\(\)](#)”을 참조하십시오.

site.editColumns()**설명**

이 함수는 [사이트 정의] 대화 상자를 표시하여 [파일 보기 열] 섹션을 보여 줍니다.

인수

없음

반환값

없음

site.exportSite()

지원 버전

Dreamweaver MX, Dreamweaver CS4에서 업데이트됨

설명

Dreamweaver 사이트를 XML 파일로 내보내며, 이 파일을 다른 Dreamweaver 인스턴스로 가져와 이전 사이트를 복제할 수 있습니다.

[사이트 정의] 대화 상자에 포함된 모든 정보가 XML 파일에 저장됩니다. 이 파일에는 숨김 폴더 목록 및 기본 문서 형식에 대한 정보가 포함됩니다. 예외적으로, FTP 액세스가 설정된 경우 사용자 로그인 및 암호를 생략할 수 있습니다.

인수

siteName, {askAboutLoginInfo}, {warnAboutSCS}, {savePath}

- siteName 인수는 내보낼 사이트를 나타냅니다. siteName이 빈 문자열인 경우 Dreamweaver에서 현재 사이트를 내보냅니다.
- askAboutLoginInfo 인수는 로그인 정보를 저장할지 묻는 대화 상자가 사용자에게 표시되는지 여부를 지정합니다. 이 인수는 선택 사항입니다.
- warnAboutSCS 인수는 소스 제어를 통해 사이트에 액세스하는 경우 로그인 정보가 저장되지 않는다는 경고가 사용자에게 표시되는지 여부를 지정합니다. 이 인수는 선택 사항입니다.
- savePath 인수는 폴더의 로컬 경로(예: C:\sites\mySites\)\입니다. savePath를 제공하는 경우 .ste 파일은 항상 사이트의 이름으로 저장됩니다. 이 인수는 선택 사항입니다.

반환값

부울 값을 반환합니다. 해당 이름의 사이트가 존재하고 XML 파일이 성공적으로 내보내졌으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

다음 예제에서는 사용자가 사이트를 내보낼 때 만들어지는 샘플 XML 파일을 보여 줍니다.

```
<?xml version="1.0" ?>
<site>
  <localinfo
    sitename="DW00"
    localroot="C:\Documents and Settings\jllondon\Desktop\DWServer\"
    imagefolder="C:\Documents and Settings\jllondon\Desktop\DWServer\Images\"
    spacerfilepath=""
    refreshlocal="TRUE"
    cache="FALSE"
    httpaddress="http://" curserver="webserver" />
  <remoteinfo
    accesstype="ftp"
    host="dreamweaver"
    remoteroot="kojak/"
    user="dream"
    checkoutname="Jay"
    emailaddress="jay@Adobe.com"
    usefirewall="FALSE"
    usepasv="TRUE"
    enablecheckin="TRUE"
    checkoutwhenopen="TRUE" />
  <designnotes
    usedesignnotes="TRUE"
    sharedesignnotes="TRUE" />
  <sitemap
    homepage="C:\Documents and Settings\jllondon\Desktop\DWServer\Untitled-2.htm"
    pagesperrow="200" columnwidth="125" showdependentfiles="TRUE"
    showpagetitles="FALSE" showhiddenfiles="TRUE" />
  <fileviewcolumns sharecolumns="TRUE">
    <column name="Local Folder"
      align="left" show="TRUE" share="FALSE" builtin="TRUE"
      localwidth="180" remotewidth="180" />
    <column name="Notes"
      align="center" show="TRUE" share="FALSE" builtin="TRUE"
      localwidth="36" remotewidth="36" />
    <column name="Size"
      align="right" show="TRUE" share="FALSE" builtin="TRUE"
      localwidth="-2" remotewidth="-2" />
    <column name="Type"
      align="left" show="TRUE" share="FALSE" builtin="TRUE"
      localwidth="60" remotewidth="60" />
    <column name="Modified"
      align="left" show="TRUE" share="FALSE" builtin="TRUE"
      localwidth="102" remotewidth="102" />
    <column name="Checked Out By"
      align="left" show="TRUE" share="FALSE" builtin="TRUE"
      localwidth="50" remotewidth="50" />
    <column name="Status" note="status"
      align="left" show="TRUE" share="FALSE" builtin="FALSE"
      localwidth="50" remotewidth="50" />
  </fileviewcolumns>
  <appserverinfo
```

```

servermodel="ColdFusion"
urlprefix="http://dreamweaver/kojak/"
serverscripting="CFML"
serverpageext=""
connectionsigrated="TRUE"
useUD4andUD5pages="TRUE"
defaultdoctype=""
accesstype="ftp"
host="dreamweaver"
remoteroot="kojak/"
user="dream"
usefirewall="FALSE"
usepasv="TRUE" />
<cloaking enabled="TRUE" patterns="TRUE">
  <cloakedfolder folder="databases/" />
  <cloakedpattern pattern=".png" />
  <cloakedpattern pattern=".jpg" />
  <cloakedpattern pattern=".jpeg" />
</cloaking>
</site>

```

site.get()

지원 버전

Dreamweaver 3

설명

다음 중 한 가지 방법으로 지정된 파일을 가져오고 종속 파일을 처리합니다.

- 사용자가 [사이트 FTP] 환경 설정에서 [내려받기/체크 아웃할 때 확인]을 선택한 경우 [종속 파일] 대화 상자가 나타납니다.
- 사용자가 이전에 [종속 파일] 대화 상자에서 [다시 나타내지 않음] 옵션을 선택하고 [예]를 클릭한 경우에는 대화 상자가 나타나지 않고 종속 파일이 다운로드됩니다.
- 사용자가 이전에 [종속 파일] 대화 상자에서 [다시 나타내지 않음] 옵션을 선택하고 [아니오]를 클릭한 경우에는 대화 상자도 나타나지 않고 종속 파일도 다운로드되지 않습니다.

인수

siteOrURL

- **siteOrURL** 인수는 "site" 키워드여야 하며, 이는 함수가 단일 파일의 URL 또는 [파일] 패널의 선택 영역에 적용되어야 함을 나타냅니다.

반환값

없음

활성자

461페이지의 “[site.canGet\(\)](#)”을 참조하십시오.

site.getAppServerAccessType()

지원 버전

Dreamweaver MX

설명

현재 사이트의 응용 프로그램 서버에 있는 모든 파일에 사용되는 액세스 방식을 반환합니다. 현재 사이트는 현재 포커스가 있는 문서와 연관된 사이트입니다. 포커스가 있는 문서가 없는 경우 Dreamweaver에서 연 사이트가 사용됩니다.

참고: ColdFusion Component Explorer에서 이 함수를 사용합니다. 212페이지의 “[site.getAppServerPathToFiles\(\)](#)” 및 215페이지의 “[site.getLocalPathToFiles\(\)](#)”를 참조하십시오.

인수

없음

반환값

다음 문자열 중 하나를 반환합니다.

- "none"
- "local/network"
- "ftp"
- "source_control"

site.getAppServerPathToFiles()**지원 버전**

Dreamweaver MX

설명

현재 사이트에 대해 정의된 응용 프로그램 서버에 있는 원격 파일의 경로를 확인합니다. 현재 사이트는 현재 포커스가 있는 문서와 연관된 사이트입니다. 포커스가 있는 문서가 없는 경우 Dreamweaver에서 연 사이트가 사용됩니다.

참고: ColdFusion Component Explorer에서 이 함수를 사용합니다. 211페이지의 “[site.getAppServerAccessType\(\)](#)” 및 215페이지의 “[site.getLocalPathToFiles\(\)](#)”를 참조하십시오.

인수

없음

반환값

응용 프로그램 서버 파일에 대한 액세스 유형이 local/network이면 경로를 반환하고 그렇지 않으면 빈 문자열을 반환합니다.

site.getAppURLPrefixForSite()**지원 버전**

Dreamweaver MX

설명

[사이트 정의] 대화 상자의 [로컬 정보] 섹션에 정의된 HTTP 주소에서 추출된 URL 접두어의 값을 가져옵니다. 이 값은 http://hostname:portnumber/ 뒤에 표시되는 경로입니다.

인수

{siteName}

선택적인 **siteName** 인수는 URL 접두어를 가져올 사이트의 이름입니다. 사이트를 지정하지 않으면 이 함수는 현재 사이트에 대한 URL 접두어를 가져옵니다.

반환값

현재 선택한 사이트의 URL 접두어를 포함하는 문자열을 반환합니다.

예제

```
var sitePrefix = site.getAppURLPrefixForSite();
```

site.getCheckoutUser()

지원 버전

Dreamweaver 3

설명

현재 사이트와 연관된 로그인과 체크 아웃 이름을 가져옵니다.

인수

없음

반환값

로그인 및 체크 아웃 이름이 정의되었으면 이들이 포함된 문자열을 반환하고 체크 인/체크 아웃이 비활성화되었으면 빈 문자열을 반환합니다.

예제

site.getCheckoutUser()를 호출하면 "denise (deniseLaptop)"가 반환될 수 있습니다. 체크 아웃 이름을 지정하지 않으면 로그인 이름만 반환됩니다(예: "denise").

site.getCheckoutUserForFile()

지원 버전

Dreamweaver 3

설명

지정한 파일을 체크 아웃한 사용자의 로그인 및 체크 아웃 이름을 가져옵니다.

인수

fileName

- **fileName** 인수는 쿼리되는 파일의 경로이며, file:// URL로 표시됩니다.

반환값

파일을 체크 아웃한 사용자의 로그인 및 체크 아웃 이름이 포함된 문자열을 반환하거나, 파일이 체크 아웃 되지 않은 경우에는 빈 문자열을 반환합니다.

예제

site.checkOutUserForFile("file:///C:/sites/avocado8/index.html")을 호출하면 "denise (deniseLaptop)"가 반환될 수 있습니다. 체크 아웃 이름을 지정하지 않으면 로그인 이름만 반환됩니다(예: "denise").

site.getCloakingEnabled()

지원 버전

Dreamweaver MX

설명

현재 사이트에 숨김이 활성화되어 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 사이트에 숨김이 활성화되어 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

site.getConnectionState()

지원 버전

Dreamweaver 3

설명

현재 연결 상태를 가져옵니다.

인수

없음

반환값

원격 사이트가 연결되었는지 여부를 나타내는 부울 값을 반환합니다.

활성자

461 페이지의 “[site.canConnect\(\)](#)”를 참조하십시오.

site.getCurrentSite()

지원 버전

Dreamweaver 3

설명

현재 사이트를 가져옵니다.

인수

없음

반환값

현재 사이트의 이름을 포함하는 문자열을 반환합니다.

예제

여러 사이트를 정의한 경우 `site.getCurrentSite()`를 호출하면 [파일] 패널의 [현재 사이트 목록]에 현재 표시되는 사이트가 반환됩니다.

site.getFocus()

지원 버전

Dreamweaver 3, CS4에서 업데이트됨

설명

포커스가 있는 [파일] 패널의 창을 확인합니다.

인수

없음

반환값

다음 로컬 문자열 또는 원격 문자열 중 하나를 반환합니다.

site.getLocalPathToFiles()

지원 버전

Dreamweaver MX

설명

현재 사이트에 대해 정의된 로컬 파일의 경로를 확인합니다. 현재 사이트는 현재 포커스가 있는 문서와 연관된 사이트입니다. 포커스가 있는 문서가 없는 경우 Dreamweaver에서 연 사이트가 사용됩니다.

참고: ColdFusion Component Explorer에서 이 함수를 사용합니다. 211페이지의 “[site.getAppServerAccessType\(\)](#)” 및 212페이지의 “[site.getAppServerPathToFiles\(\)](#)”를 참조하십시오.

인수

없음

반환값

현재 사이트의 로컬 컴퓨터에 있는 파일의 경로를 반환합니다.

site.getLocalRootURL()

지원 버전

Dreamweaver CS4

설명

사이트의 로컬 루트 폴더를 가져옵니다.

인수*siteName*

- *siteName* 인수는 사이트의 이름을 지정하는 문자열입니다.

반환값

해당 이름의 사이트에 대한 로컬 루트 폴더를 포함하는 문자열을 반환합니다. 이 폴더는 **file://URL**로 표시됩니다. 지정된 사이트가 없는 경우에는 빈 문자열을 반환합니다.

site.getSelection()**지원 버전**

Dreamweaver 3

설명

[파일] 패널에서 현재 선택된 파일을 확인합니다.

인수

없음

반환값

선택된 파일과 폴더의 경로를 나타내는 **file:// URL**로 표시되는 문자열의 배열을 반환하거나, 파일이나 폴더가 선택되지 않은 경우에는 빈 배열을 반환합니다.

site.getSiteForURL()**지원 버전**

Dreamweaver MX

설명

특정 파일과 연관된 사이트가 있는 경우 해당 사이트의 이름을 가져옵니다.

인수*fileURL*

- *fileURL* 인수는 이름이 지정된 파일의 **"file://"** 문자열을 포함하는 전체 URL입니다.

반환값

지정된 파일이 존재하는 사이트가 있으면 해당 사이트의 이름을 포함하는 문자열을 반환하고 지정된 파일이 정의된 사이트에 없으면 빈 문자열을 반환합니다.

site.getSites()**지원 버전**

Dreamweaver 3

설명

정의된 사이트의 목록을 가져옵니다.

인수

없음

반환값

정의된 사이트의 이름을 나타내는 문자열의 배열을 반환하거나, 정의된 사이트가 없는 경우에는 빈 배열을 반환합니다.

site.getSiteRootForURL()

지원 버전

Dreamweaver CS4

설명

특정 파일 URL과 연관된 사이트의 로컬 루트 폴더를 가져옵니다.

인수

fileURL

- fileURL 인수는 이름이 지정된 파일의 "file://" 문자열을 포함하는 전체 URL이 포함된 문자열 인수입니다.

반환값

지정한 파일이 있는 사이트의 로컬 루트 폴더를 포함하는 문자열을 반환합니다. 이 폴더는 file://URL로 표시됩니다. 지정한 파일이 정의된 사이트에 없으면 빈 문자열을 반환합니다.

예제

```
var dom = dw.getDocumentDOM();  
var siteRoot = site.getSiteRootForURL(dom.URL);
```

site.getSiteURLPrefix()

지원 버전

Dreamweaver 8

설명

[로컬 정보] 섹션에 정의된 HTTP 주소에서 추출된 사이트 URL 접두어를 가져옵니다.

인수

없음

반환값

사이트 URL 접두어를 포함하는 문자열을 반환합니다.

예제

```
sitePrefix = getSiteURLPrefix();
```

site.importSite()

지원 버전

Dreamweaver MX

설명

XML 파일에서 Dreamweaver 사이트를 생성합니다. Dreamweaver에서는 <localinfo> 요소의 localroot 속성을 사용하여 사이트의 로컬 루트 폴더를 식별합니다. 가져오는 동안 이 폴더가 로컬 컴퓨터에 없으면 다른 로컬 루트 폴더를 묻는 메시지가 표시됩니다. Dreamweaver에서는 <localinfo> 요소의 imagefolder 속성으로 지정한 기본 이미지 폴더를 찾으려고 할 때도 동일한 방식으로 동작합니다.

인수

pathToSteFile

- pathToSteFile 인수는 STE 파일의 URL을 포함하는 문자열입니다. Dreamweaver에서는 이 파일을 사용하여 사이트를 생성합니다. **pathToSteFile**이 빈 문자열인 경우 가져올 STE 파일을 선택하도록 요청하는 메시지가 표시됩니다.

반환값

부울 값을 반환합니다. 해당 이름의 STE 파일이 있고 사이트가 성공적으로 만들어졌으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

site.isCloaked()

지원 버전

Dreamweaver MX

설명

[파일] 패널의 현재 선택 영역이나 지정된 폴더의 숨김 여부를 확인합니다.

인수

siteOrURL

- **siteOrURL** 인수는 다음 두 값 중 하나를 포함해야 합니다.
 - "site" 키워드 - isCloaked() 함수가 [파일] 패널의 선택 영역을 테스트해야 함을 나타냅니다..
 - 특정 폴더의 파일 URL - isCloaked()가 지정된 폴더를 테스트해야 함을 나타냅니다.

반환값

부울 값을 반환합니다. 지정된 객체가 숨겨졌으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

site.locateInSite()

지원 버전

Dreamweaver 3

설명

[파일] 패널의 지정된 창에서 지정된 파일을 찾아 선택합니다.

인수**localOrRemote, siteOrURL**

- **localOrRemote** 인수는 "local" 또는 "remote"여야 합니다.
- **siteOrURL** 인수는 "site" 키워드여야 하며, 이는 함수가 단일 파일의 URL 또는 [파일] 패널의 선택 영역에 적용되어야 함을 나타냅니다.

반환값

없음

활성자462페이지의 “[site.canLocateInSite\(\)](#)”를 참조하십시오.**site.makeEditable()****지원 버전**

Dreamweaver 3

설명

선택된 파일에서 읽기 전용 플래그를 해제합니다.

인수

없음

반환값

없음

활성자462페이지의 “[site.canMakeEditable\(\)](#)”을 참조하십시오.**site.makeNewDreamweaverFile()****지원 버전**

Dreamweaver 3

설명

[파일] 패널에서 선택된 첫 번째 파일이나 폴더와 동일한 폴더에 새 Dreamweaver 파일을 만듭니다.

인수

없음

반환값

없음

활성자463페이지의 “[site.canMakeNewFileOrFolder\(\)](#)”를 참조하십시오.

site.makeNewFolder()

지원 버전

Dreamweaver 3

설명

[파일] 패널에서 선택된 첫 번째 파일이나 폴더와 동일한 폴더에 새 폴더를 만듭니다.

인수

없음

반환값

없음

활성자

463페이지의 “[site.canMakeNewFileOrFolder\(\)](#)”를 참조하십시오.

site.newSite()

지원 버전

Dreamweaver 3

설명

새로운 무제 사이트의 [사이트 정의] 대화 상자를 엽니다.

인수

없음

반환값

없음

site.open()

지원 버전

Dreamweaver 3

설명

[파일] 패널에서 현재 선택된 파일을 엽니다. 폴더가 선택된 경우 [사이트 파일] 뷰에서 확장됩니다.

인수

없음

반환값

없음

활성자

463페이지의 “[site.canOpen\(\)](#)”을 참조하십시오.

site.put()**지원 버전**

Dreamweaver 3

설명

다음 방식 중 하나로 선택된 파일을 올리고 종속 파일을 처리합니다.

- 사용자가 [사이트 FTP] 환경 설정에서 [올리기/체크 인할 때 확인]을 선택한 경우 [종속 파일] 대화 상자가 나타납니다.
- 사용자가 이전에 [종속 파일] 대화 상자에서 [다시 나타내지 않음] 옵션을 선택하고 [예]를 클릭한 경우에는 대화 상자가 나타나지 않고 종속 파일이 업로드됩니다.
- 사용자가 이전에 [종속 파일] 대화 상자에서 [다시 나타내지 않음] 옵션을 선택하고 [아니오]를 클릭한 경우에는 대화 상자도 나타나지 않고 종속 파일도 업로드되지 않습니다.

인수**siteOrURL**

- **siteOrURL** 인수는 "site" 키워드여야 하며, 이는 함수가 단일 파일의 URL 또는 [파일] 패널의 선택 영역에 적용되어야 함을 나타냅니다.

반환값

없음

활성자

463페이지의 “[site.canPut\(\)](#)”을 참조하십시오.

site.recreateCache()**지원 버전**

Dreamweaver 3

설명

현재 사이트의 캐시를 다시 만듭니다.

인수

없음

반환값

없음

활성자

464페이지의 “[site.canRecreateCache\(\)](#)”를 참조하십시오.

site.refresh()

지원 버전

Dreamweaver 3, CS4에서 업데이트됨

설명

[파일] 패널의 지정된 면에 있는 파일 목록을 새로 고칩니다.

인수

whichSide

- whichSide 인수는 local 또는 remote여야 합니다.

반환값

없음

활성자

464페이지의 “[site.canRefresh\(\)](#)”를 참조하십시오.

site.remotelsValid()

지원 버전

Dreamweaver 3

설명

원격 사이트가 유효한지 여부를 확인합니다.

인수

없음

반환값

원격 사이트가 정의되었는지 여부와 서버 유형이 로컬/네트워크인 경우 드라이브가 마운트되었는지 여부를 나타내는 부울 값을 반환합니다.

site.renameSelection()

지원 버전

Dreamweaver 3

설명

선택된 파일의 이름을 텍스트 필드로 전환하여 사용자가 파일의 이름을 새로 지정할 수 있도록 합니다. 둘 이상의 파일을 선택하면 이 함수는 마지막으로 선택된 파일에 적용됩니다.

인수

없음

반환값

없음

site.runValidation()

지원 버전

Dreamweaver MX

설명

유효성 검사기를 전체 사이트에 대해 또는 강조된 항목에 대해서만 실행합니다.

인수

selection

- **selection** 인수는 유효성 검사기가 강조 표시된 항목만 확인하도록 지정하는 매개 변수입니다. 이 인수를 지정하지 않으면 유효성 검사기가 현재 사이트 전체를 확인합니다.

반환값

없음

site.selectAll()

지원 버전

Dreamweaver 3, CS4에서 업데이트됨

설명

활성 뷰에서 모든 파일을 선택합니다.

인수

없음

반환값

없음

site.selectNewer()

지원 버전

Dreamweaver 3

설명

[파일] 패널의 지정된 면에서 최근의 파일을 모두 선택합니다.

인수

whichSide

- **whichSide** 인수는 "local" 또는 "remote"여야 합니다.

반환값

없음

활성자465페이지의 “[site.canSelectNewer\(\)](#)”를 참조하십시오.

site.serverActivity()

지원 버전

Dreamweaver 8

설명

이 함수는 Dreamweaver가 서버와 상호 작용하고 있는지 여부를 확인합니다. Dreamweaver에서는 한 번에 둘 이상의 서버 작업을 수행할 수 없으므로 서버 상호 작용이 필요한 기능을 비활성화할지 여부를 결정할 때 이 함수를 사용할 수 있습니다.

인수

없음

반환값

Dreamweaver가 서버와 상호 작용하고 있는지 여부를 나타내는 부울 값을 반환합니다.

예제

menus.xml 파일의 다음 예제에서는 서버 작업이 없을 경우(및 Dreamweaver에 현재 사이트가 지정되어 있을 경우) 메뉴 항목을 표시합니다.

```
<menuitem name="Remove Connection Scripts" enabled="!site.serverActivity() &&
site.getCurrentSite() != ''" command="alert(MMDB.removeConnectionScripts())"
id="SiteOptionsSiteMenu_RemoveConnectionScripts" />
```

site.setCloakingEnabled()

지원 버전

Dreamweaver MX

설명

현재 사이트에 숨김을 활성화할지 여부를 결정합니다.

인수**enable**

- **enable** 인수는 숨김을 활성화할지 여부를 나타내는 부울 값입니다. true 값은 현재 사이트에서 숨김을 활성화하고, false 값은 현재 사이트에서 숨김을 비활성화합니다.

반환값

없음

site.setConnectionState()

지원 버전

Dreamweaver 3

설명

현재 사이트의 연결 상태를 설정합니다.

인수

bConnected

- **bConnected** 인수는 현재 사이트와 연결되어 있는지(true), 아니면 연결되어 있지 않은지(false)를 나타내는 부울 값입니다.

반환값

없음

site.setCurrentSite()

지원 버전

Dreamweaver 3

설명

[파일] 패널의 로컬 창에서 지정된 사이트를 엽니다.

인수

whichSite

- **whichSite** 인수는 정의된 사이트의 이름입니다. 이 이름은 [파일] 패널이나 [사이트 정의] 대화 상자의 [현재 사이트] 목록에 나타납니다.

반환값

없음

예제

세 개의 사이트가 정의된 경우(예: avocado8, dreamcentral 및 testsite), site.setCurrentSite("dreamcentral")를 호출하면 dreamcentral이 현재 사이트가 됩니다.

site.setFocus()

지원 버전

Dreamweaver 3, CS4에서 업데이트됨

설명

[파일] 패널의 지정된 창에 포커스를 둡니다. 지정된 창이 표시되지 않은 경우 해당 창을 표시하고 포커스를 둡니다.

인수

whichPane, nextTextView

- **whichPane** 인수는 local 또는 remote 문자열 중 하나여야 합니다.
- nextTextView 인수는 분할 뷰의 뷰 간에 포커스를 전환합니다.

반환값

없음

site.setSelection()

지원 버전

Dreamweaver 3

설명

[파일] 패널의 활성화된 창에서 파일 또는 폴더를 선택합니다.

인수

arrayOfURLs

- arrayOfURLs 인수는 file:// URL로 표시되는 문자열의 배열로, 각 문자열은 현재 사이트의 파일 또는 폴더 경로입니다.

참고: 폴더 경로를 지정할 때는 뒤에 오는 슬래시(/)를 생략합니다.

반환값

없음

site.siteRelativeToLocalPath()

지원 버전

Dreamweaver 8

설명

사이트 상대 URI 참조를 로컬 파일 경로로 변환합니다.

인수

siteRelativeURI

- 필수적인 siteRelativeURI 속성은 사이트 상대 URI를 포함하는 문자열입니다.

반환값

로컬 컴퓨터의 로컬 파일에 대한 경로를 지정하는 문자열을 반환합니다.

예제

다음 예제에서는

```
var filePath = site.siteRelativeToLocalPath("/myWebApp/myFile.xml");
```

[사이트 정의] 대화 상자의 [로컬 정보]에 지정된 사이트 매핑과 HTTP 주소에 따라 "C:\inetpub\wwwroot\siteA\myFile.xml"을 반환합니다.

site.synchronize()

지원 버전

Dreamweaver 3

설명

[파일 동기화] 대화 상자를 엽니다.

인수

없음

반환값

없음

활성자

465페이지의 “[site.canSynchronize\(\)](#)”를 참조하십시오.

site.uncloak()

지원 버전

Dreamweaver MX

설명

[파일] 패널의 현재 선택 영역이나 지정된 폴더를 표시합니다.

인수

siteOrURL

- **siteOrURL** 인수는 다음 값 중 하나를 포함해야 합니다.
 - "site" 키워드 - unCloak() 함수가 [파일] 패널의 선택 영역에 적용되어야 함을 나타냅니다..
 - 특정 폴더의 URL - unCloak() 함수가 지정된 폴더와 해당 폴더의 모든 내용에 적용되어야 함을 나타냅니다.

반환값

없음

활성자

465페이지의 “[site.canUncloak\(\)](#)”을 참조하십시오.

site.uncloakAll()

지원 버전

Dreamweaver MX

설명

현재 사이트의 모든 폴더를 표시하며 [숨김] 설정에서 [다음으로 끝나는 파일 숨김] 체크 상자의 선택을 해제합니다.

인수

없음

반환값

없음

활성자

465페이지의 “[site.canUncloak\(\)](#)”을 참조하십시오.

site.undoCheckOut()

지원 버전

Dreamweaver 3

설명

지정된 파일과 연관된 잠금 파일을 로컬 및 원격 사이트에서 제거하고, 지정된 파일의 로컬 사본을 원격 사본으로 바꿉니다.

인수

siteOrURL

- **siteOrURL** 인수는 "site" 키워드여야 하며, 이는 함수가 단일 파일의 URL 또는 [파일] 패널의 선택 영역에 적용되어야 함을 나타냅니다.

반환값

없음

활성자

466페이지의 “[site.canUndoCheckOut\(\)](#)”을 참조하십시오.

14장: 문서

Adobe® Dreamweaver®의 문서 함수는 사용자가 작업하고 있는 문서에 적용되는 작업을 수행합니다. 문서 함수를 사용하여 다음과 같은 작업을 수행할 수 있습니다.

- 표를 레이어로 변환
- Configuration/Commands 폴더에서 명령 실행
- 파일 URL 탐색
- 상대 URL을 절대 URL로 변환
- 현재 선택된 노트 가져오기
- 문자열에서 URL 인코딩 수행
- 문서에서 변환기 실행

변환 함수

변환 함수는 표를 레이어로, 레이어를 표로, CSS(Cascading Style Sheet)를 HTML 마크업으로 변환합니다. 각 함수는 [파일] 또는 [수정] 메뉴에 있는 변환 명령 중 하나의 비헤이비어를 정확하게 복제합니다.

dom.convertLayersToTable()

지원 버전

Dreamweaver 3

설명

[레이어를 표로 변환] 대화 상자를 엽니다.

인수

없음

반환값

없음

활성자

433페이지의 “[dom.canConvertLayersToTable\(\)](#)”을 참조하십시오.

dom.convertTablesToLayers()

지원 버전

Dreamweaver 3

설명

[표를 레이어로 변환] 대화 상자를 엽니다.

인수

없음

반환값

없음

활성자

433페이지의 “[dom.canConvertTablesToLayers\(\)](#)”를 참조하십시오.

명령 함수

명령 함수를 사용하면 Configuration/Commands 폴더에 대부분의 파일을 만들 수 있습니다. 이러한 함수는 [명령] 메뉴를 관리하고 다른 유형의 확장 파일에서 명령을 호출합니다.

dreamweaver.editCommandList()

지원 버전

Dreamweaver 3

설명

[명령 목록 편집] 대화 상자를 엽니다.

인수

없음

반환값

없음

dreamweaver.runCommand()

지원 버전

Dreamweaver 3

설명

지정된 명령을 실행합니다. 메뉴에서 명령을 선택하는 작업과 같습니다. 대화 상자가 명령과 연관되어 있으면 대화 상자가 나타나고 명령 스크립트는 사용자가 대화 상자를 닫을 때까지 다른 편집 작업을 차단합니다. 이 함수는 또 다른 확장 파일에서 명령을 호출할 수 있습니다.

참고: 이 함수는 objectTag() 함수 내부, 명령 파일의 스크립트 또는 속성 관리자 파일에서 호출할 수 있습니다.

인수

`commandFile`, {`commandArg1`}, {`commandArg2`},...{`commandArgN`}

- **commandFile** 인수는 Configuration/Commands 폴더 내의 파일 이름입니다.
- 나머지 인수 **commandArg1**, **commandArg2** 등은 선택적이며 **commandFile** 인수의 `receiveArguments()` 함수에 전달됩니다.

반환값

없음

예제

버튼의 `onClick` 이벤트 핸들러에서 다음 함수를 호출하여 사용자가 속성 관리자의 버튼을 통해 [표 형태 만들기] 명령을 사용할 수 있도록 사용자 정의 표 속성 관리자를 작성할 수 있습니다.

```
function callFormatTable() {
    dreamweaver.runCommand('Format Table.htm');
}
```

파일 조작 함수

파일 조작 함수는 XML 및 XHTML 등의 문서를 만들고 열고 저장하며, 기존 HTML 문서를 XHTML로 변환하고, CSS를 외부 파일로 내보내는 등의 작업을 처리합니다. 이러한 함수는 파일 또는 폴더 검색, 템플릿 기반 파일 만들기, 문서 닫기, 최근에 열린 본 파일에 대한 정보 가져오기 등의 작업을 수행합니다.

dom.cleanupXHTML()

지원 버전

Dreamweaver MX

설명

이 함수는 `convertToXHTML()` 함수와 비슷하지만 기존 XHTML 문서를 정리합니다. 이 함수는 문서 내의 선택 영역에 대해 실행될 수 있습니다. `cleanupXHTML()` 함수를 실행하면 전체 XHTML 문서 또는 문서의 현재 선택 영역에서 구문을 정리할 수 있습니다.

인수

bWholeDoc

- **bWholeDoc** 인수는 부울 값을 갖습니다. 값이 `true`이면 `cleanupXHTML()` 함수가 전체 문서를 정리하고 그렇지 않으면 선택 영역만 정리합니다.

반환값

6개의 정수로 이루어진 배열을 반환합니다. 각 정수는 다음 요소의 수를 나타냅니다.

- Dreamweaver에서 수정한 XHTML 오류
- id 속성이 없고 수정할 수 없는 map 요소
- type 속성이 없고 수정할 수 없는 script 요소
- type 속성이 없고 수정할 수 없는 style 요소

- alt 속성이 없고 수정할 수 없는 img 요소
- alt 속성이 없고 수정할 수 없는 area 요소

dom.convertToXHTML()

지원 버전

Dreamweaver MX

설명

HTML을 DOM 트리 구조로 파싱하고 XHTML에 필요한 누락된 항목을 삽입한 다음 트리를 정리하고 완전한 XHTML로 트리를 작성합니다. convertToXHTML() 함수가 DOM 트리에 필요에 따라 추가할 수 있는 누락된 지시문, 선언, 요소 및 속성은 다음과 같습니다.

- XML 지시문
- doctype 선언
- html 요소의 xmlns 속성
- head 섹션
- title 요소
- body 섹션

변환 과정에서 dom.convertToXHTML() 함수는 순수 HTML 태그와 속성을 소문자로 변환하고 HTML 태그와 속성을 올바른 XHTML 구문으로 작성하며 누락된 HTML 속성을 해당 위치에 추가합니다. 이 함수는 [환경 설정] 대화 상자의 설정에 따라 타사 태그 및 속성을 처리합니다.

문서가 템플릿인 경우 dom.convertToXHTML() 함수는 사용자에게 경고하지만 변환을 수행하지는 않습니다.

인수

없음

반환값

6개의 정수로 이루어진 배열을 반환합니다. 각 정수는 다음 요소의 수를 나타냅니다.

- Dreamweaver에서 수정한 XHTML 오류
- id 속성이 없고 수정할 수 없는 map 요소
- type 속성이 없고 수정할 수 없는 script 요소
- type 속성이 없고 수정할 수 없는 style 요소
- alt 속성이 없고 수정할 수 없는 img 요소
- alt 속성이 없고 수정할 수 없는 area 요소

예제

일반적인 사용의 경우 Extension은 먼저 dreamweaver.openDocument() 또는 dreamweaver.getDocumentDOM() 함수를 호출하여 문서에 대한 참조를 가져옵니다. 그런 다음 dom.getIsXHTMLDocument() 함수를 호출하여 문서가 이미 XHTML 양식인지 여부를 확인합니다. 문서가 XHTML 양식이 아니면 Extension은 dom.convertToXHTML() 함수를 호출하여 문서를 XHTML로 변환합니다. 그런 다음 dreamweaver.saveDocument() 함수를 호출하여 변환된 파일을 새 파일 이름으로 저장합니다.

dom.getIsXHTMLDocument()

지원 버전

Dreamweaver MX

설명

문서(특히 <!DOCTYPE> 선언)를 검사하여 XHTML인지 여부를 확인합니다.

인수

없음

반환값

문서가 XHTML이면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.browseForFileURL()

지원 버전

Dreamweaver 1, Dreamweaver 2, 3 및 4에서 기능이 향상됨

설명

제목 표시줄에 지정된 레이블이 있는 지정된 유형의 대화 상자를 엽니다.

인수

openSelectOrSave, {*titleBarLabel*}, {*bShowPreviewPane*}, {*bSupressSiteRootWarnings*}, {*arrayOfExtensions*}, {*startFolder*}, {*allowDynamic*}, {*fileToLocate*}

- *openSelectOrSave* 인수는 대화 상자의 유형을 *open*, *select* 또는 *save*로 나타내는 문자열입니다.
- *titleBarLabel* 인수(Dreamweaver 2에서 추가됨)는 대화 상자의 제목 표시줄에 표시되는 레이블입니다. 이 인수가 생략되는 경우 운영 체제에서 제공하는 기본 레이블이 사용됩니다.
- *bShowPreviewPane* 인수(Dreamweaver 2에서 추가됨)는 대화 상자에 [이미지 미리 보기] 창을 표시할지 여부를 나타내는 부울 값입니다. 이 인수의 값이 true이면 대화 상자에서 이미지 파일을 필터링하며, 이 인수가 생략되는 경우 기본값은 false입니다.
- *bSupressSiteRootWarnings* 인수(Dreamweaver 3에서 추가됨)는 선택한 파일이 사이트 루트 외부에 있는 경우 이를 경고하는 메시지를 숨길지 여부를 나타내는 부울 값입니다. 이 인수가 생략되는 경우 기본값은 false입니다.
- *arrayOfExtensions* 인수(Dreamweaver 4에서 추가됨)는 문자열의 배열로, 대화 상자의 아래쪽에 있는 파일 유형 목록 메뉴의 기본 내용을 지정합니다. 이 인수의 구문은 `menuEntryText[.xxx[:.yyy:.zzz]][CCCC]`이며, 여기서
 - *menuEntryText*는 파일 유형의 이름입니다.
 - 확장명을 `.xxx[:.yyy:.zzz]` 또는 `CCCC`로 지정할 수 있습니다.
 - `.xxx`는 파일 유형의 파일 이름 확장명을 지정합니다. 파일 이름 확장명을 여러 개 지정하려면 `.yyy` 및 `.zzz`를 사용합니다.
 - `CCCC`는 Macintosh에서 사용되는 네 자리의 파일 유형 상수입니다.

다음 예제에서는 [선택] 대화 상자에서 MP3 파일에 대한 필터와 모든 파일에 대한 필터를 제공합니다.

```
dw.browseForFileURL("select", "Please select an mp3",false, true, new Array("mp3 Files (*.MP3)|*.mp3||", "All Files (*.*)|*.*||"));
```

- **startFolder** 인수는 검색이 시작되는 폴더의 파일 URL을 지정하는 데 사용할 수 있는 문자열 값입니다. 이 인수가 지정되지 않은 경우 검색이 마지막으로 사용된 디렉토리에서 시작됩니다. 이 인수는 선택 사항입니다.
- **allowDynamic** 인수는 동적 URL이나 매개 변수를 허용할지 여부를 나타내는 부울 값입니다. 이 인수의 값이 **true**이면 동적 URL이나 매개 변수가 허용됩니다. 이 인수는 선택 사항입니다.
- **fileToLocate** 인수는 찾을 파일의 파일 URL을 지정하는 데 사용되는 문자열 값입니다. 이 인수는 선택 사항입니다.

반환값

file://URL로 표시되는 파일 이름을 포함하는 문자열을 반환합니다.

dreamweaver.browseForFolderURL()

지원 버전

Dreamweaver 3

설명

제목 표시줄에 지정된 레이블이 있는 [폴더 선택] 대화 상자를 엽니다.

인수

{titleBarLabel}, {directoryToStartIn}

- **titleBarLabel** 인수는 대화 상자의 제목 표시줄에 표시될 레이블입니다. 이 인수가 생략되는 경우 **titleBarLabel** 인수의 기본값은 [폴더 선택]입니다.
- **directoryToStartIn** 인수는 폴더를 열 경로이며, file:// URL로 표시됩니다.

반환값

file:// URL로 표시되는 폴더 이름을 포함하는 문자열을 반환합니다.

예제

다음 코드에서는 폴더의 URL을 반환합니다.

```
return dreamweaver.browseForFolderURL('Select a Folder', ~
dreamweaver.getSiteRoot());
```

dreamweaver.closeDocument()

지원 버전

Dreamweaver 2

설명

지정된 문서를 닫습니다.

인수

documentObject

- **documentObject** 인수는 문서의 DOM 트리 루트에 있는 객체(dreamweaver.getDocumentDOM() 함수가 반환하는 값)입니다. **documentObject** 인수가 활성 문서를 참조하는 경우에는 이 함수를 호출하는 스크립트의 실행이 완료될 때까지 [문서] 윈도우가 닫히지 않을 수 있습니다.

반환값
없음

dreamweaver.createDocument()

지원 버전

Dreamweaver 2, Dreamweaver 4에서 기능이 향상됨

설명

이 함수는 전달되는 인수에 따라 새 문서를 같은 윈도우나 새 윈도우에서 엽니다. 새 문서가 활성 문서가 됩니다.

참고: 이 함수는 `menus.xml` 파일, 명령 또는 속성 관리자 파일에서만 호출할 수 있습니다. 비헤이비어 액션이나 객체에서 이 함수를 호출하려고 하면 오류 메시지가 표시됩니다.

인수

{bOpenInSameWindow}, {type}

- **bOpenInSameWindow** 인수는 새 문서를 현재 윈도우에서 열지 여부를 나타내는 부울 값입니다.
bOpenInSameWindow 인수가 `false`이거나 생략되는 경우 또는 이 함수가 Macintosh에서 호출되는 경우 새 문서는 별도의 윈도우에서 열립니다.
- **type** 인수는 Dreamweaver Configuration/DocumentTypes/MMDocumentTypes.xml 파일에서 documenttype 태그의 id 속성으로 선언된 대로 만들 문서의 유형을 지정합니다. 예를 들어, **type** 인수는 "HTML", "ASP-JS", "ASP-VB", "ColdFusion", "CFC", "JSP", "ASP.NET_VB" 등이 될 수 있습니다. 가능한 유형의 전체 목록은 MMDocumentTypes.xml 파일을 참조하십시오. **type**을 지정하지 않으면 기본적으로 "HTML" 값이 지정됩니다.

참고: 사용자 고유의 문서 형식을 추가하여 MMDocumentTypes 파일을 확장할 수 있습니다. 문서 형식 확장에 대한 자세한 내용은 Dreamweaver 확장을 참조하십시오.

반환값

새로 만들어진 문서에 대한 문서 객체를 반환합니다. 이 값은 `dreamweaver.getDocumentDOM()` 함수가 반환하는 값과 같습니다.

dreamweaver.createXHTMLDocument()

지원 버전

Dreamweaver MX

설명

이 함수는 전달되는 인수에 따라 새 XHTML 문서를 같은 윈도우나 새 윈도우에서 엽니다. 새 문서가 활성 문서가 됩니다. 이 함수는 `dreamweaver.createDocument()` 함수와 유사합니다.

Dreamweaver에서는 새 XHTML 문서를 만들 때 Configurations/Templates 폴더에 있는 `default.xhtml`이라는 파일을 읽고 해당 파일의 내용을 사용하여 다음과 같은 기본 선언을 포함하는 출력 파일을 만듭니다.

```
<?xml version="1.0">
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=" />
</head>

<body bgcolor="#FFFFFF" text="#000000">

</body>
</html>
```

기본 DTD(문서 형식 정의) 선언은 Strict가 아닌 XHTML 1.0 Transitional입니다. 사용자가 문서에 프레임세트를 추가하면 DTD가 XHTML 1.0 Frameset로 전환됩니다. Content-Type은 text/html이고, charset은 의도적으로 default.xhtml 파일의 값이 적용되지 않고 유지되지만 사용자가 새 문서를 볼 때는 이 값이 채워집니다. 문서에서 UTF-8이나 UTF-16 문자 인코딩을 사용하는 경우에는 ?xml 지시문이 필요하지 않습니다. 이 지시문이 있으면 이전 브라우저에서 렌더링될 수 있습니다. 그러나 XHTML 문서에는 이 지시문이 있어야 하므로 기본적으로 새 문서와 변환된 문서 모두에 대해 이 지시문이 사용됩니다. 필요한 경우 사용자가 수동으로 이 지시문을 삭제할 수 있습니다. ?xml 지시문에는 Content-Type 속성의 charset과 일치하는 인코딩 속성이 포함됩니다.

인수

{bOpenInSameWindow}

- **bOpenInSameWindow** 인수는 새 문서를 현재 윈도우에서 열지 여부를 나타내는 부울 값입니다. 이 값이 false이거나 생략되는 경우 또는 이 함수가 Macintosh에서 호출되는 경우 새 문서는 별도의 윈도우에서 열립니다.

반환값

새로 만들어진 문서에 대한 문서 객체를 반환합니다. 이 값은 dreamweaver.getDocumentDOM() 함수가 반환하는 값과 같습니다.

dreamweaver.createXMLDocument()

지원 버전

Dreamweaver MX

설명

XML 지시문만 포함하는 새 XML 파일을 만들어 엽니다.

인수

없음

반환값

새 XML 파일의 DOM을 반환합니다.

예제

다음 예제에서는 XML 지시문만 포함된 새 문서를 만듭니다.

```
var theDOM = dreamweaver.createXMLDocument("document");
```

dreamweaver.exportTemplateDataAsXML()

지원 버전

Dreamweaver MX

설명

현재 문서를 지정된 파일에 XML로 내보냅니다. 이 함수는 포커스가 있는 문서에 작동하며, 문서는 반드시 템플릿이어야 합니다. 파일 이름 인수를 지정하지 않으면 내보내기 파일의 문자열을 입력할 수 있는 대화 상자가 열립니다.

인수

{filePath}

- 선택적인 **filePath** 인수는 템플릿을 내보낼 대상 파일 이름을 지정하는 문자열입니다. **filePath** 인수를 "file:///c:/temp/mydata.txt"와 같은 URL 파일 문자열로 표시해야 합니다.

반환값

없음

활성자

443페이지의 "[dreamweaver.canExportTemplateDataAsXML\(\)](#)"을 참조하십시오.

예제

```
if (dreamweaver.canExportTemplateDataAsXML())  
{  
    dreamweaver.exportTemplateDataAsXML("file:///c:/dw_temps/mytemplate.txt")  
}
```

dreamweaver.getDocumentDOM()

지원 버전

Dreamweaver 2

설명

지정된 문서의 객체 트리에 액세스할 수 있게 합니다. 객체 트리가 호출자에 반환되면 호출자는 트리를 편집하여 문서의 내용을 변경할 수 있습니다.

인수

{sourceDoc}

- sourceDoc** 인수는 "document", "parent", "parent.frames[number]", "parent.frames[frameName]" 또는 URL이어야 합니다. sourceDoc 값을 제공하지 않는 경우 기본값은 "document"입니다. 이러한 인수 값의 의미는 다음과 같습니다.
 - document 값은 포커스가 있고 현재 선택 영역을 포함하는 문서를 지정합니다.
 - parent 값은 현재 선택한 문서가 프레임에 있는 경우 부모 프레임셋을 지정합니다.
 - parent.frames[number] 및 parent.frames[frameName] 값은 현재 문서를 포함하는 프레임셋 내의 특정 프레임에 있는 문서를 지정합니다.
- 인수가 상대 URL이면 확장 파일에 대해 상대적입니다.

참고: 인수가 "document"이면 호출 함수는 `applyBehavior()`, `deleteBehavior()`, `objectTag()` 함수 또는 문서 편집을 수행할 수 있는 명령이나 속성 관리자 파일의 함수여야 합니다.

반환값

트리의 루트에 있는 JavaScript 문서 객체를 반환합니다.

예제

다음 예제에서는 `dreamweaver.getDocumentDOM()` 함수를 사용하여 현재 문서에 액세스합니다.

```
var theDOM = dreamweaver.getDocumentDOM("document");
```

다음 예제에서는 현재 문서 DOM으로 선택 영역을 식별하여 다른 문서의 끝에 붙여 넣습니다.

```
var currentDOM = dreamweaver.getDocumentDOM('document');
currentDOM.setSelection(100,200);
currentDOM.clipCopy();
var otherDOM = dreamweaver.openDocument(dreamweaver.
getSiteRoot() + "html/foo.htm");
otherDOM.endOfDocument();
otherDOM.clipPaste();
```

참고: DOM 메서드는 열려 있는 문서에서만 정상적으로 작동하기 때문에 `openDocument()` 인수가 사용됩니다. 열려 있지 않은 문서에서 함수를 실행하면 Dreamweaver 오류가 발생합니다. DOM 메서드의 설명을 보면 활성 문서에서만 작동하는지, 아니면 닫혀 있는 문서에서도 작동할 수 있는지를 알 수 있습니다.

dreamweaver.getNewDocumentDOM()

지원 버전

Dreamweaver MX, Dreamweaver 8에 `documentType` 인수가 추가됨

설명

비어 있는 새 문서의 편집 가능한 트리에 액세스할 수 있도록 합니다. 이 함수는 기존 문서가 아닌 새 문서를 가리키고 문서를 열지 않는다는 점을 제외하고 `getDocumentDOM()` 함수와 같은 방식으로 작동합니다.

인수

{`documentType`}

- **documentType** 인수는 값이 `DocumentTypes.xml` 파일에 지정된 문서 형식이어야 하는 문자열입니다.

반환값

비어 있는 새 문서에 대한 포인터를 반환합니다.

예제

다음 코드에서는 비어 있는 새 문서에 대한 DOM을 반환합니다.

```
var theDOM = dreamweaver.getNewDocumentDOM();
```

dreamweaver.getRecentFileList()

지원 버전

Dreamweaver 3

설명

[파일] 메뉴 아래쪽에 있는 최근에 사용한 파일 목록의 모든 파일 목록을 가져옵니다.

인수

없음

반환값

가장 최근에 액세스한 파일들의 경로를 나타내는 문자열 배열을 반환합니다. 각 경로는 file:// URL로 표시됩니다. 최근에 사용한 파일이 없으면 반환되는 값이 없습니다.

dreamweaver.importXMLIntoTemplate()

지원 버전

Dreamweaver 3

설명

XML 텍스트 파일을 현재 템플릿 문서로 가져옵니다. 이 함수는 포커스가 있는 문서에 작동하며, 문서는 반드시 템플릿이어야 합니다. 파일 이름 인수를 지정하지 않으면 Dreamweaver에서 가져오기 파일의 문자열을 입력할 수 있는 대화 상자가 열립니다.

인수

{filePath}

- 선택적인 **filePath** 인수는 템플릿을 가져올 대상 파일 이름을 지정하는 문자열입니다. **filePath** 인수를 "file:///c:/temp/mydata.txt"와 같은 URL 파일 문자열로 표시해야 합니다.

반환값

없음

dreamweaver.newDocument()

지원 버전

Dreamweaver MX

설명

현재 사이트에서 문서를 열고 [새 문서] 대화 상자를 시작합니다.

인수

{bopenWithCurSiteAndShowDialog}

- 선택적인 **bopenWithCurSiteAndShowDialog** 인수의 값은 true 또는 false입니다. 현재 사이트에 문서를 열고 [새 문서] 대화 상자를 표시하려면 true로 지정하고 그렇지 않으면 false로 지정합니다.

반환값

없음

dreamweaver.newFromTemplate()

지원 버전

Dreamweaver 3

설명

지정된 템플릿에서 새 문서를 만듭니다. 인수를 제공하지 않으면 [템플릿 선택] 대화 상자가 나타납니다.

인수

{templateURL}, bMaintain

- **templateURL** 인수는 현재 사이트에 있는 템플릿의 경로이며, file:// URL로 표시됩니다.
- **bMaintain** 인수는 원본 템플릿에 대한 링크를 유지할지 여부를 나타내는 부울 값으로, true 또는 false입니다.

반환값

없음

dreamweaver.openDocument()

지원 버전

Dreamweaver 2

설명

새 Dreamweaver 윈도우에서 편집할 문서를 열고 포커스를 둡니다. 사용자 입장에서는 [파일] > [열기]를 선택하고 파일을 선택하는 것과 같습니다. 지정된 파일이 이미 열려 있으면 해당 문서를 포함하는 윈도우가 맨 앞으로 옵니다. 지정된 파일을 포함하는 윈도우가 현재 선택된 윈도우가 됩니다. Dreamweaver 2에서는 체크 인/체크 아웃이 활성화된 경우 파일이 열리기 전에 체크 아웃됩니다. Dreamweaver 3 이상에서는 240페이지의 “[dreamweaver.openDocumentFromSite\(\)](#)”를 사용하여 이 비헤이비어를 가져와야 합니다.

참고: 이 함수를 비헤이비어 액션 또는 객체 파일에서 호출하면 오류가 발생합니다.

인수

fileName

- **fileName** 인수는 열 파일의 이름이며, URL로 표시됩니다. URL이 상대적이면 이 함수를 호출한 스크립트가 포함된 파일에 대해 상대적입니다.

반환값

지정된 파일의 문서 객체를 반환합니다. 이 값은 dreamweaver.getDocumentDOM() 함수가 반환하는 값과 같습니다.

dreamweaver.openDocumentFromSite()

지원 버전

Dreamweaver 3

설명

새 Dreamweaver 윈도우에서 편집할 문서를 열고 포커스를 둡니다. 사용자 입장에서는 [사이트] 패널에서 파일을 두 번 클릭하는 것과 같습니다. 지정된 파일이 이미 열려 있으면 해당 문서를 포함하는 윈도우가 맨 앞으로 옵니다. 지정된 파일을 포함하는 윈도우가 현재 선택된 윈도우가 됩니다.

참고: 오류가 발생하기 때문에 비헤이비어 액션이나 객체 파일에서는 이 함수를 호출할 수 없습니다.

인수**fileName**

- **fileName** 인수는 열 파일이며, URL로 표시됩니다. URL이 상대적이면 이 함수를 호출한 스크립트가 포함된 파일에 대해 상대적입니다.

반환값

지정된 파일의 문서 객체를 반환합니다. 이 값은 `dreamweaver.getDocumentDOM()` 함수가 반환하는 값과 같습니다.

dreamweaver.openInFrame()

지원 버전

Dreamweaver 3

설명

[프레임에서 열기] 대화 상자를 엽니다. 사용자가 문서를 선택하면 문서가 활성 프레임에서 열립니다.

인수

없음

반환값

없음

활성자

444페이지의 “[dreamweaver.canOpenInFrame\(\)](#)”을 참조하십시오.

dreamweaver.releaseDocument()

지원 버전

Dreamweaver 2

설명

이전에 참조한 문서를 메모리에서 명시적으로 해제합니다.

`dreamweaver.getObjectTags()`, `dreamweaver.getObjectRefs()`, `dreamweaver.getDocumentPath()` 또는 `dreamweaver.getDocumentDOM()` 함수가 참조하는 문서는 해당 호출을 포함하는 스크립트 실행이 완료되면 자동으로 해제됩니다. 스크립트에서 많은 문서를 여는 경우에는 스크립트가 완료되기 전에 이 함수를 사용하여 명시적으로 문서를 해제해야 메모리 부족 현상을 방지할 수 있습니다.

참고: 이 함수는 URL에 의해 참조되고 현재 프레임이나 문서 윈도우에서 열려 있지 않으며 확장 파일이 아닌 문서에만 해당됩니다. 확장 파일은 Dreamweaver를 시작할 때 메모리에 로드되며 Dreamweaver를 종료할 때까지 해제되지 않습니다.

인수**documentObject**

- **documentObject** 인수는 문서의 DOM 트리 루트에 있는 객체(`dreamweaver.getDocumentDOM()` 함수가 반환하는 값)입니다.

반환값

없음

dreamweaver.revertDocument()

지원 버전

Dreamweaver 3

설명

지정된 문서를 이전에 저장된 버전으로 되돌립니다.

인수

`documentObject`, `warn`

- **documentObject** 인수는 문서의 DOM 트리 루트에 있는 객체(`dreamweaver.getDocumentDOM()` 함수가 반환하는 값)입니다.
- **warn** 인수는 저장하지 않은 변경 내용이 적용되지 않는다고 사용자에게 경고할지 여부를 지정하는 부울 값입니다. 이 인수가 제공되지 않는 경우 기본값은 `true`입니다.

반환값

부울 값을 반환합니다. Dreamweaver에서 사용자에게 경고해야 하면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

활성자

446페이지의 “[dreamweaver.canRevertDocument\(\)](#)”를 참조하십시오.

dreamweaver.saveAll()

지원 버전

Dreamweaver 3

설명

열려 있는 모든 문서를 저장하며, 이전에 저장되지 않은 문서의 경우 [다른 이름으로 저장] 대화 상자를 엽니다.

인수

없음

반환값

없음

활성자

446페이지의 “[dreamweaver.canSaveAll\(\)](#)”을 참조하십시오.

dreamweaver.saveDocument()

지원 버전

Dreamweaver 2

설명

지정된 파일을 로컬 컴퓨터에 저장합니다.

참고: Dreamweaver 2에서는 파일이 읽기 전용이면 파일의 체크 아웃이 시도됩니다. 이러한 시도 후에도 문서가 계속 읽기 전용이거나 문서를 만들 수 없으면 오류 메시지가 나타납니다.

인수

documentObject, {fileURL}

- **documentObject** 인수는 문서의 DOM 트리 루트에 있는 객체(dreamweaver.getDocumentDOM() 함수가 반환하는 값)입니다.
- 선택적인 **fileURL** 인수는 로컬 컴퓨터에서의 위치를 나타내는 URL입니다. URL이 상대적이면 확장 파일에 대해 상대적입니다. Dreamweaver 2에서는 이 인수가 필수적입니다. Dreamweaver 4에서 **fileURL** 인수가 생략된 경우에는 파일이 이전에 저장되었으면 현재 위치에 저장되고 그렇지 않으면 [저장] 대화 상자가 나타납니다.

반환값

성공(true) 또는 실패(false)를 나타내는 부울 값을 반환합니다.

활성자

447페이지의 “[dreamweaver.canSaveDocument\(\)](#)”를 참조하십시오.

dreamweaver.saveDocumentAs()

지원 버전

Dreamweaver 3

설명

[다른 이름으로 저장] 대화 상자를 엽니다.

인수

documentObject

- **documentObject** 인수는 문서의 DOM 트리 루트에 있는 객체(dreamweaver.getDocumentDOM() 함수가 반환하는 값)입니다.

반환값

없음

dreamweaver.saveDocumentAsTemplate()

지원 버전

Dreamweaver 3

설명

[템플릿으로 저장] 대화 상자를 엽니다.

인수

documentObject, {fileName}

- **documentObject** 인수는 문서의 DOM 트리 루트에 있는 객체(`dreamweaver.getDocumentDOM()`이 반환하는 값)입니다.
- 선택적인 **fileName** 인수는 열 파일의 이름이며, 절대 URL로 표시됩니다.

반환값

없음

활성자

447페이지의 “[dreamweaver.canSaveDocumentAsTemplate\(\)](#)”를 참조하십시오.

dreamweaver.saveFrameset()

지원 버전

Dreamweaver 3

설명

지정된 프레임세트를 저장하거나, 프레임세트가 이전에 저장되지 않은 경우 [다른 이름으로 저장] 대화 상자를 엽니다.

인수

documentObject

- **documentObject** 인수는 문서의 DOM 트리 루트에 있는 객체(`dreamweaver.getDocumentDOM()` 함수가 반환하는 값)입니다.

반환값

없음

활성자

447페이지의 “[dreamweaver.canSaveFrameset\(\)](#)”를 참조하십시오.

dreamweaver.saveFramesetAs()

지원 버전

Dreamweaver 3

설명

지정된 DOM을 포함하는 프레임세트 파일에 대해 [다른 이름으로 저장] 대화 상자를 엽니다.

인수

documentObject

- **documentObject** 인수는 문서의 DOM 트리 루트에 있는 객체(`dreamweaver.getDocumentDOM()` 함수가 반환하는 값)입니다.

반환값

없음

활성자

448페이지의 “[dreamweaver.canSaveFramesetAs\(\)](#)”를 참조하십시오.

전역 문서 함수

전역 문서 함수는 전체 문서에 대해 작용합니다. 맞춤법 검사, 대상 브라우저 확인, 페이지 속성 설정, 문서에 있는 요소의 올바른 객체 참조 확인 등과 같은 작업을 수행합니다.

dom.checkSpelling()

지원 버전

Dreamweaver 3

설명

문서의 맞춤법을 검사하고 필요한 경우 [맞춤법 검사] 대화 상자를 열어 검사가 완료되면 사용자에게 알려 줍니다.

인수

없음

반환값

없음

dom.checkTargetBrowsers()

지원 버전

Dreamweaver 3

설명

문서에서 대상 브라우저 확인 작업을 실행합니다. 폴더나 파일 그룹에서 대상 브라우저 확인 작업을 실행하려면 205페이지의 “[site.checkTargetBrowsers\(\)](#)”를 참조하십시오.

인수

없음

반환값

없음

dom.getParseMode()

지원 버전

Dreamweaver MX 2004

설명

문서의 유효성 검사 방법과 문서가 기본 문서 윈도우에 HTML로 표시되는지 여부를 제어하는 문서의 현재 파싱 모드를 가져옵니다.

인수

없음

반환값

현재 파싱 모드 "html", "xml", "css" 또는 "text"를 지정하는 문자열을 반환합니다.

dom.hideInfoMessagePopup()

지원 버전

Dreamweaver MX 2004

설명

문서 윈도우에 도구 설명과 비슷한 메시지가 표시되면 이를 숨깁니다.

인수

없음

반환값

없음

참고 항목

247페이지의 “[dom.showInfoMessagePopup\(\)](#)”

dom.runValidation()

지원 버전

Dreamweaver MX, Dreamweaver MX 2004에서 선택적인 인수가 추가됨

설명

지정된 하나의 문서에서 유효성 검사를 실행합니다. 이 함수는 223페이지의 “[site.runValidation\(\)](#)”과 유사합니다. 유효성 검사는 문서가 문서 **doctype**에 지정된 언어(예: HTML 4.0 또는 HTML 3.2) 및 서버 모델로 지정된 언어(예: ColdFusion 또는 ASP) 규칙을 따르는지 검사합니다. 문서에 **doctype**이 없으면 유효성 검사는 [환경 설정] 대화 상자의 [유효성 검사기] 섹션에 지정된 언어 설정을 사용합니다.

인수

{controlString}, {bOpenResultsWindow}, {bShowInfoMessage}

- **controlString** 인수는 선택적 문자열로, 빈 문자열, "xml", "auto-explicit" 또는 "auto-implicit" 중 하나일 수 있습니다.
- 인수가 빈 문자열이면 유효성 검사는 기본 유효성 검사를 수행합니다. 인수가 "xml"이면 유효성 검사는 XML 기반 방식으로 문서 유효성을 검사합니다.

- 인수가 "auto-explicit" 또는 "auto-implicit"이면 자동 유효성 검사(인라인 유효성 검사라고도 함)가 수행됩니다. 이 유효성 검사에서는 [유효성 검사] 결과 윈도우를 여는 대신 [코드] 뷰에서 오류에 밑줄을 그어 표시합니다(417페이지의 “[dom.source.getValidationErrorsForOffset\(\)](#)” 및 411페이지의 “[dom.getAutoValidationCount\(\)](#)” 참조).
- **controlString** 인수가 "auto-explicit"이면 유효성 검사가 실행되기 전에 저장되지 않은 문서를 저장할지 묻는 메시지가 표시됩니다.
- **controlString** 인수가 "auto-implicit"이면 현재 문서가 저장되지 않았다는 것을 사용자에게 알리지 않고 유효성 검사가 실패합니다.

참고: 자동 유효성 검사(**controlString** 값 "auto-explicit" 또는 "auto-implicit"으로 정의)는 현재 브라우저 호환성 확인에서만 사용할 수 있습니다.

- **bOpenResultsWindow** 인수는 선택적 부울 값입니다. 값이 true이면 [유효성 검사] 결과 윈도우를 열고, 값이 false이면 열지 않습니다. 기본값은 true입니다.
- **bShowInfoMessage** 인수는 **controlString** 인수가 "auto-explicit" 또는 "auto-implicit"으로 정의된 경우에만 사용됩니다. **bShowInfoMessage** 인수는 부울 값입니다. 값이 true이면 툴바 항목 아래에 발견된 오류 수와 함께 DW_ValidatorErrors라는 정보 메시지가 표시되고, 값이 false이면 아무 것도 표시되지 않습니다. 기본값은 false입니다.

반환값

[유효성 검사] 결과 윈도우 객체를 반환합니다.

예제

다음 예제에서는 사용자가 [파일] > [페이지 확인] > [마크업 유효성 검사] 메뉴 옵션(또는 [유효성 검사] 패널의 [현재 문서 유효성 검사])을 선택할 때 일반 유효성 검사를 실행합니다.

```
dw.getDocumentDOM().runValidation('');
```

다음 예제에서는 저장되지 않은 문서를 저장할지 묻는 메시지를 표시하고 자동 유효성 검사를 실행한 다음 [유효성 검사] 결과 윈도우를 열지 않고 DW_ValidatorErrors의 문서 툴바 버튼에 총 오류 수를 보여 줍니다.

```
dw.getDocumentDOM().runValidation('auto-explicit', false, true);
```

다음 예제에서는 저장되지 않은 문서를 저장할지 묻는 메시지를 표시하지 않습니다. 문서가 저장되어 있지 않으면 유효성 검사가 시작되지 않습니다. 문서가 저장되어 있으면 자동 유효성 검사가 실행됩니다. 이 경우 [유효성 검사] 결과 윈도우가 열리지 않으며 문서 툴바에 발생한 총 오류 수가 표시되지도 않습니다.

```
dw.getDocumentDOM().runValidation('auto-implicit', false);
```

dom.showInfoMessagePopup()

지원 버전

Dreamweaver MX 2004

설명

문서 윈도우 안이나 툴바 항목 아래에 도구 설명과 비슷한 메시지를 표시합니다.

인수

location, message, timeout

- **location** 인수는 툴바 항목을 지정하는 문자열, 빈 문자열 또는 "top", "topright", "right", "bottomright", "bottom", "bottomleft", "left" 또는 "topleft" 키워드 중 하나입니다. 도구 설명은 지정된 가장자리 또는 모서리에 가운데 정렬됩니다. 빈 문자열을 지정하면 문서의 가운데에 위치하게 됩니다. 툴바 항목을 지정하려면 "toolbar:toolbarID:itemID"를 사용합니다. 여기서 툴바 ID와 툴바 항목 ID는 toolbars.xml 파일의 ID와 일치합니다.

- **message** 인수는 메시지를 포함하는 문자열입니다.
- **timeout** 인수는 메시지를 표시할 밀리초를 지정하는 숫자입니다. 기본값은 0입니다. 값이 0이면 메시지가 계속 표시됩니다. 사용자가 메시지를 클릭하거나 문서를 전환하는 경우 또는 시간이 초과되는 경우 자동으로 메시지가 닫힙니다.

반환값

없음

예제

다음 예제에서는 두 개의 도구 설명 메시지를 표시합니다. 첫 번째 코드 행은 문서의 가운데에 "This message is in the center"라는 메시지를 표시합니다. 두 번째 `showInfoMessagePopup()` 호출은 ID가 `DW_Toolbar_Main`인 톨바에 있는 ID가 `DW_SetTitle`인 [제목] 텍스트 편집 상자에 도구 설명 메시지 "Don't forget the title for the Window"를 표시합니다.

```
dw.getDocumentDOM.showInfoMessagePopup('', 'This message is in the center', 5000);  
dw.getDocumentDOM.showInfoMessagePopup('toolbar:DW_Toolbar_Main:DW_SetTitle', 'Don't  
forget the title for the window', 5000);
```

참고 항목

246페이지의 “[dom.hideInfoMessagePopup\(\)](#)”

dom.showPagePropertiesDialog()

지원 버전

Dreamweaver 3

설명

[페이지 속성] 대화 상자를 엽니다.

인수

없음

반환값

없음

dreamweaver.doURLDecoding()

지원 버전

Dreamweaver MX

설명

내부 Dreamweaver URL 디코딩 메커니즘을 사용하여 URL 문자열의 특수 문자와 기호를 디코딩합니다. 예를 들어, 이 함수는 `%20`을 공백 문자로 디코딩하고, `"`를 `"`로 디코딩합니다.

인수

`inStr`

- `inStr` 인수는 디코딩할 문자열입니다.

반환값

디코딩된 URL이 포함된 문자열을 반환합니다.

예제

다음 예제에서는 `dw.doURLDecoding()`을 호출하여 해당 인수의 특수 문자를 디코딩하고 결과 문자열을 `outstr`에 저장합니다.

```
outStr = dreamweaver.doURLDecoding("http://maps.yahoo.com/py/ddResults.py?Pyt= ↵
Tmap&tarname=&tardesc=&newname=&newdesc=&newHash=&newTHash=&newSts=&newTSts=&tlr=&tlr= ↵
&slt=&sln=&newFL=Use+Address+Below&newaddr=2000+Shamrock+Rd&newcsz=Metroo+Park%2C+CA& ↵
newcountry=us&newTFL=Use+Address+Below&newtaddr=500+El+Camino&newtcsz=Santa+Clara%2C+CA& ↵
newcountry=us&Submit=Get+Directions")
```

dreamweaver.getElementRef()

지원 버전

Dreamweaver 2

설명

DOM 트리에 있는 특정 태그 객체의 Netscape Navigator 또는 Internet Explorer 객체 참조를 가져옵니다.

인수

NSorIE, tagObject

- **NSorIE** 인수는 "NS 4.0" 또는 "IE 4.0"이어야 합니다. 중첩된 참조의 DOM 및 규칙은 Netscape Navigator 4.0과 Internet Explorer 4.0에서 다릅니다. 이 인수는 유효한 참조를 반환할 브라우저를 지정합니다.
- **tagObject** 인수는 DOM 트리의 태그 객체입니다.

반환값

`document.layers['myLayer']`와 같이 객체에 대한 유효한 JavaScript 참조를 나타내는 문자열을 반환합니다. 문자열은 다음 조건에 따라 결정됩니다.

- Dreamweaver에서 A, AREA, APPLET, EMBED, DIV, SPAN, INPUT, SELECT, OPTION, TEXTAREA, OBJECT 및 IMG 태그에 대한 Internet Explorer의 올바른 참조를 반환합니다.
- Dreamweaver에서 A, AREA, APPLET, EMBED, LAYER, ILAYER, SELECT, OPTION, TEXTAREA, OBJECT 및 IMG 태그와 절대 위치가 지정된 DIV 및 SPAN 태그에 대한 Netscape Navigator의 올바른 참조를 반환합니다. 절대 위치가 지정되지 않은 DIV 및 SPAN 태그에 대해서는 "cannot reference <tag>"를 반환합니다.
- Dreamweaver에서 이름이 없는 객체에 대한 참조를 반환하지 않습니다. 객체에 NAME 또는 ID 속성이 포함되지 않은 경우 Dreamweaver에서 "unnamed <tag>"를 반환합니다. 브라우저가 이름으로 참조하는 것을 지원하지 않으면 Dreamweaver에서 인덱스로 객체를 참조합니다(예: `document.myform.applets[3]`).
- Dreamweaver에서 이름이 없는 양식과 레이아웃에 포함된 이름이 있는 객체에 대한 참조를 반환합니다(예: `document.forms[2].myCheckbox`).

dreamweaver.getPreferenceInt()

지원 버전

Dreamweaver MX

설명

Extension에 대한 환경 설정(정수로 표시됨)을 가져올 수 있습니다.

인수

section, key, default_value

- **section** 인수는 항목이 포함된 환경 설정 섹션을 지정하는 문자열입니다.
- **key** 인수는 가져올 값의 항목을 지정하는 문자열입니다.
- **default_value** 인수는 항목을 찾을 수 없는 경우 Dreamweaver에서 반환하는 기본값입니다. 이 값은 0에서 65,535 사이의 부호 없는 정수이거나 -32,768에서 32,767 사이의 부호 있는 정수여야 합니다.

반환값

지정된 섹션에 있는 지정된 항목의 정수 값을 반환하거나, 항목을 찾을 수 없는 경우에는 기본값을 반환합니다. 지정된 항목의 값이 정수가 아닌 경우에는 0을 반환합니다.

예제

다음 예제에서는 [환경 설정]의 [내 Extension] 섹션에 있는 [물리기 거리] 설정의 정수 값을 반환합니다. [내 Extension] 섹션 또는 [물리기 거리] 항목이 없으면 함수는 지정된 기본값 0을 반환합니다.

```
var snapDist; //default value if entry not found
snapDist = dreamweaver.getPreferenceInt("My Extension", "Snap Distance", 0);
```

dreamweaver.getPreferenceString()

지원 버전

Dreamweaver MX

참고: 사이트 환경 설정에 액세스하려면 버전이 7.0.1이어야 합니다. 사이트 정보에 액세스하기 전에 dw.appVersion에서 버전이 올바른지 확인하십시오.

설명

Extension에 대해 저장한 환경 설정(문자열로 표시됨)을 가져올 수 있습니다.

인수

section, key, default_value

- **section** 인수는 항목이 포함된 환경 설정 섹션을 지정하는 문자열입니다.
- **key** 인수는 가져올 값을 지정하는 문자열입니다.
- **default_value** 인수는 항목을 찾을 수 없는 경우 Dreamweaver에서 반환하는 기본 문자열 값입니다.

반환값

요청한 환경 설정 문자열을 반환하거나, 문자열을 찾을 수 없는 경우에는 기본값을 반환합니다.

예제

다음 예제에서는 [환경 설정]의 [내 Extension] 섹션에 있는 [텍스트 편집기] 설정의 문자열 값을 반환합니다. [내 Extension] 섹션 또는 [텍스트 편집기] 항목이 없으면 함수는 txtEditor 변수로 지정된 기본값을 반환합니다.

```
var txtEditor = getExternalTextEditor(); //set default text Editor value
txtEditor = dreamweaver.getPreferenceString("My Extension", "Text Editor", txtEditor);
```

dreamweaver.setPreferenceInt()

지원 버전

Dreamweaver MX

설명

Extension에 대한 환경 설정(정수로 표시됨)을 설정할 수 있습니다. 이 설정은 Dreamweaver가 실행되고 있지 않을 때 Dreamweaver 환경 설정에 저장됩니다.

인수

section, key, new_value

- **section** 인수는 옵션이 설정되는 환경 설정 범주를 지정하는 문자열입니다. 범주가 없는 경우 Dreamweaver에서 항목을 만듭니다.
- **key** 인수는 함수가 설정하는 범주 옵션을 지정하는 문자열입니다. 옵션이 없는 경우 Dreamweaver에서 항목을 만듭니다.
- **new_value** 인수는 범주 옵션의 값을 지정하는 정수입니다.

반환값

성공하면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

다음 예제에서는 [물리기 거리] 항목을 [환경 설정]의 [내 Extension] 범주에 있는 snapDist 변수의 값으로 설정합니다.

```
var snapDist = getSnapDistance();
if(snapDist > 0)
{
    dreamweaver.setPreferenceInt("My Extension", "Snap Distance", snapDist);
}
```

dreamweaver.setPreferenceString()

지원 버전

Dreamweaver MX

참고: 사이트 환경 설정에 액세스하려면 버전이 7.0.1이어야 합니다. 사이트 정보에 액세스하기 전에 dw.appVersion에서 버전이 올바른지 확인하십시오.

설명

Extension에 대한 환경 설정(문자열로 표시됨)을 작성할 수 있습니다. 이 설정은 Dreamweaver가 실행되고 있지 않을 때 Dreamweaver 환경 설정에 저장됩니다.

인수

section, key, new_value

- **section** 인수는 옵션이 설정되는 [환경 설정] 범주를 지정하는 문자열입니다. 범주가 없는 경우 Dreamweaver에서 항목을 만듭니다.
- **key** 인수는 함수가 설정하는 범주 옵션을 지정하는 문자열입니다. 범주 옵션이 없는 경우 Dreamweaver에서 항목을 만듭니다.
- **new_value** 인수는 범주 옵션의 값을 지정하는 문자열입니다.

반환값

성공하면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

```
var txtEditor = getExternalTextEditor();  
dreamweaver.setPreferenceString("My Extension", "Text Editor", txtEditor);
```

dreamweaver.showTargetBrowsersDialog()

지원 버전

Dreamweaver MX 2004

설명

[대상 브라우저] 대화 상자를 엽니다. [대상 브라우저] 대화 상자에서는 사용자가 현재 페이지의 브라우저 호환성 문제를 검사하기 위해 [브라우저 대상 확인] 기능에서 사용할 브라우저 버전을 지정할 수 있습니다.

인수

없음

반환값

없음

경로 함수

경로 함수는 사용자의 하드 디스크에 있는 다양한 파일과 폴더의 경로를 가져오고 조작합니다. 이러한 함수는 현재 문서가 있는 사이트 루트의 경로를 확인하고 상대 경로를 절대 URL로 변환하는 등의 작업을 수행합니다.

dreamweaver.absoluteURLToDocRelative()

지원 버전

Dreamweaver 2

설명

절대 URL과 문서의 경로가 주어지면 이 함수는 절대 URL을 문서에 상대적인 경로로 변환합니다.

인수

docPathURL, siteRootURL, absoluteURL

- docPathURL 인수는 사용자 컴퓨터에 있는 문서(예: 현재 문서)의 경로이며 file://URL로 표시됩니다.
- siteRootURL 인수는 사이트 루트의 경로이며 file://URL로 표시됩니다.
- absoluteURL 인수는 문서에 상대적인 경로로 변환할 file://URL입니다.

반환값

absoluteURL에 있는 문서의 경로를 나타내는 문자열을 반환하며, docPathURL에 있는 문서에 대해 상대적으로 표시됩니다.

예제

다음 예제에서 **docPathURL** 및 **siteRootURL** 값이 각각 file://C:/sites/cherrystreet/archives/october.shtml 및 file://C:/sites/cherrystreet/인 경우 반환값은 "../includes/header.html"입니다. /archives/october.shtml에서 /includes/header.html을 참조하려면 이 값을 사용합니다.

```
var docPathURL = dw.getDocumentDOM().URL;
var siteRootURL = dw.getSiteRoot();
var absoluteURL= dw.relativeToAbsoluteURL(docPathURL, siteRootURL, "/includes/header.html");
var docRelPath = dw.absoluteURLToDocRelative(docPathURL, siteRootURL, absoluteURL);
```

dreamweaver.getConfigurationPath()

지원 버전

Dreamweaver 2

설명

file:// URL로 표시되는 Dreamweaver Configuration 폴더의 경로를 가져옵니다.

Dreamweaver에서 다중 사용자 플랫폼의 Configuration 폴더에 액세스하는 방법에 대한 자세한 내용은 Dreamweaver 확장 도움말에서 “C 레벨 확장성”을 참조하십시오.

인수

없음

반환값

응용 프로그램 구성의 경로를 반환합니다.

예제

다음 함수는 Dreamweaver 응용 프로그램 폴더 내의 Configuration 폴더에 저장되는 다른 확장 파일을 참조할 때 유용합니다.

```
var sortCmd = dreamweaver.getConfigurationPath() + "\n"
"/Commands/Sort Table.htm"
var sortDOM = dreamweaver.getDocumentDOM(sortCmd);
```

dreamweaver.getDocumentPath()

지원 버전

Dreamweaver 1.2

설명

file:// URL로 표시되는 지정된 문서의 경로를 가져옵니다. 이 함수는 dreamweaver.getDocumentDOM()을 호출한 다음 반환값의 URL 속성을 읽는 것과 동일한 결과를 나타냅니다.

인수

sourceDoc

- **sourceDoc** 인수의 값은 "document", "parent", "parent.frames[number]" 또는 "parent.frames[frameName]"이어야 합니다. "document" 값은 포커스가 있고 현재 선택 영역을 포함하는 문서를 지정합니다. "parent" 값은 부모 프레임셋을 지정하고(현재 선택된 문서가 프레임에 있는 경우), "parent.frames[number]" 및 "parent.frames[frameName]" 값은 현재 문서가 포함된 프레임셋 내의 특정 프레임에 있는 문서를 지정합니다.

반환값

파일이 저장되었으면 지정된 문서의 URL을 포함하는 문자열을 반환하고 그렇지 않으면 빈 문자열을 반환합니다.

dreamweaver.getSiteRoot()

지원 버전

Dreamweaver 1.2

설명

현재 선택된 문서와 연관된 사이트의 로컬 루트 폴더([사이트 정의] 대화 상자에서 지정)를 가져오며, file:// URL로 표시됩니다.

인수

없음

반환값

파일이 저장된 사이트의 로컬 루트 폴더 URL을 포함하는 문자열을 반환하거나, 파일이 사이트와 연관되지 않은 경우에는 빈 문자열을 반환합니다.

dreamweaver.getTempFolderPath()

지원 버전

Dreamweaver MX

설명

임시 파일을 저장할 수 있는 임시 폴더의 전체 경로를 가져옵니다. 이 함수는 Dreamweaver Configuration 폴더에서 Temp 폴더를 찾습니다. 시스템에서 다중 사용자를 지원하는 경우 사용자의 Configuration 폴더를 찾습니다. Temp 폴더가 없으면 새로 만들어집니다. 임시 파일이 아닌 공유 파일은 Configuration/Shared 폴더에 저장해야 합니다.

인수

없음

반환값

file:// URL로 표시되는 폴더의 전체 경로를 반환합니다.

예제

다음 코드 행은 지정된 파일의 전체 경로를 반환합니다. dw.getTempFolderPath() 함수는 다른 Dreamweaver 함수(예: dreamweaver.getSiteRoot())와 달리 경로의 끝에 슬래시(/)를 반환하지 않습니다.

```
var myTempfile = dw.getTempFolderPath() + "/myTempFile.txt";
```

dreamweaver.relativeToAbsoluteURL()

지원 버전

Dreamweaver 2

설명

상대 URL과 참조점(사이트 루트나 현재 문서의 경로)이 주어지면 이 함수는 상대 URL을 절대 file:// URL로 변환합니다.

인수

docPath, **siteRoot**, **relURL**

- **docPath** 인수는 사용자 컴퓨터에 있는 문서(예: 현재 문서)의 경로이며 file:// URL로 표시됩니다. **relURL**이 루트에 상대적인 URL인 경우 빈 문자열입니다.
- **siteRoot** 인수는 사이트 루트의 경로이며 file:// URL로 표시됩니다. **relURL**이 문서에 상대적인 URL인 경우 빈 문자열입니다.
- **relURL** 인수는 변환할 URL입니다.

반환값

절대 URL 문자열을 반환합니다. 반환값은 다음에 설명된 대로 생성됩니다.

- **relURL**이 절대 URL이면 변환되지 않으며, 반환값은 **relURL**과 동일합니다.
- **relURL**이 문서에 상대적인 URL이면 반환값은 **docPath** 및 **relURL**입니다.
- **relURL**이 루트에 상대적인 URL이면 반환값은 **siteRoot** 및 **relURL**입니다.

선택 함수

선택 함수는 열린 문서에서 선택 영역을 가져오고 설정합니다. [사이트] 패널에서 선택 영역을 가져오거나 설정하는 방법에 대한 자세한 내용은 198페이지의 “[사이트 함수](#)”를 참조하십시오.

dom.getSelectedNode()

지원 버전

Dreamweaver 3

설명

선택한 노드를 가져옵니다. 이 함수는 dom.getSelection() 함수를 호출하여 반환값을 dom.offsetsToNode() 함수에 전달하는 것과 동일한 결과를 나타냅니다.

인수

없음

반환값

지정된 범위의 문자가 모두 포함되는 태그, 텍스트 또는 주석 객체를 반환합니다.

dom.getSelection()

지원 버전

Dreamweaver 3

설명

문서 소스 코드까지의 문자 오프셋으로 표시되는 선택 영역을 가져옵니다.

인수

{bAllowMultiple}

- 선택적인 **bAllowMultiple** 인수는 둘 이상의 표 셀, 이미지 맵 핫스팟이나 레이어가 선택된 경우 함수가 여러 오프셋을 반환할지 여부를 나타내는 부울 값입니다.

이 인수가 생략되는 경우 기본값은 **false**입니다.

반환값

간단한 선택 영역의 경우 정수 두 개를 포함하는 배열을 반환합니다. 첫 번째 정수는 선택 영역 처음까지의 문자 오프셋이고 두 번째 정수는 선택 영역 끝까지의 문자 오프셋입니다. 두 숫자가 동일할 경우 현재 선택 영역은 삽입점입니다.

복잡한 선택 영역(다중 표 셀, 다중 레이어 또는 다중 이미지 맵 핫스팟)의 경우, **2n**개의 정수가 포함된 배열을 반환합니다. 여기서 **n**은 선택된 항목의 수입니다. 각 쌍의 첫 번째 정수는 열기 태그 TD, DIV, SPAN, LAYER, ILAYER 또는 MAP을 포함하는 선택 영역 처음까지의 문자 오프셋이고, 두 번째 정수는 닫기 태그 TD, DIV, SPAN, LAYER, ILAYER 또는 MAP을 포함하는 선택 영역 끝까지의 문자 오프셋입니다. 다중 표 행이 선택되면 각 행에 있는 각 셀의 오프셋이 반환됩니다. 선택 영역에는 TR 태그가 포함되지 않습니다.

dom.getSelectorsDefinedInStylesheet()

지원 버전

Dreamweaver 8

설명

속성으로 전달된 형식과 일치하는 선택기의 배열을 가져옵니다.

인수

selector

- selector 인수는 class 또는 ID 값의 문자열로, 함수에서 반환하는 선택기의 유형이 class인지, 아니면 ID인지를 지정합니다.

반환값

class 또는 ID 유형일 수 있는 선택기의 배열을 반환합니다.

예제

다음 코드는 class 유형의 선택기 배열을 가져오는 데 사용됩니다.

```
var dom=dw.getDocumentDOM();
var classSelectors = dom.getSelectorsDefinedInStylesheet('class');
```

다음 코드는 ID 유형의 선택기 배열을 가져오는 데 사용됩니다.

```
var dom=dw.getDocumentDOM();
var classSelectors = dom.getSelectorsDefinedInStylesheet('ID');
```

dom.nodeToOffsets()

지원 버전

Dreamweaver 3

설명

DOM 트리에서의 특정 노드의 위치를 가져옵니다. 이 위치는 문서 소스 코드까지의 문자 오프셋으로 표시됩니다. 이 함수는 로컬 드라이브에 있는 모든 문서에서 유효합니다.

인수

node

- **node** 인수는 `dreamweaver.getDocumentDOM()` 함수가 반환하는 트리의 노드인 태그, 주석 또는 일정 범위의 텍스트여야 합니다.

반환값

두 개의 정수를 포함하는 배열을 반환합니다. 첫 번째 정수는 태그, 텍스트 또는 주석의 시작 부분까지의 문자 오프셋이고, 두 번째 정수는 HTML 문서의 처음 부분에서 노드 끝까지의 문자 오프셋입니다.

예제

다음 코드에서는 현재 문서에서 첫 번째 이미지 객체를 선택합니다.

```
var theDOM = dw.getDocumentDOM();
var theImg = theDOM.images[0];
var offsets = theDOM.nodeToOffsets(theImg);
theDOM.setSelection(offsets[0], offsets[1]);
```

dom.offsetsToNode()

지원 버전

Dreamweaver 3

설명

지정된 시작점 및 끝점 사이의 문자가 모두 포함되는 DOM 트리의 객체를 가져옵니다. 이 함수는 로컬 드라이브에 있는 모든 문서에서 유효합니다.

인수

offsetBegin, offsetEnd

- **offsetBegin** 인수는 문서의 시작 부분에서 DOM 트리의 객체인 문자 범위의 시작 부분까지의 오프셋을 지정합니다.
- **offsetEnd** 인수는 문서의 시작 부분에서 DOM 트리의 객체인 문자 범위의 끝 부분까지의 오프셋을 지정합니다.

반환값

지정된 범위의 문자가 모두 포함되는 태그, 텍스트 또는 주석 객체를 반환합니다.

예제

다음 코드에서는 선택 영역이 이미지인 경우 경고를 표시합니다.

```
var offsets = dom.getSelection();
var theSelection = dreamweaver.offsetsToNode(offsets[0], ~
offsets[1]);
if (theSelection.nodeType == Node.ELEMENT_NODE && ~
theSelection.tagName == 'IMG'){
    alert('The current selection is an image.');
```

dom.selectAll()

지원 버전

Dreamweaver 3

설명

[모두 선택] 작업을 수행합니다.

참고: 대부분의 경우 이 함수는 활성 문서의 모든 내용을 선택합니다. 그러나 특정한 경우(예: 삽입점이 표 안에 있을 때)에는 활성 문서의 일부만 선택합니다. 선택 영역을 문서 전체로 설정하려면 `dom.setSelection()`을 사용합니다.

인수

없음

반환값

없음

dom.setSelectedNode()

지원 버전

Dreamweaver 3

설명

선택한 노드를 설정합니다. 이 함수는 `dom.nodeToOffsets()` 함수를 호출하여 반환값을 `dom.setSelection()` 함수에 전달하는 것과 동일한 결과를 나타냅니다.

인수

`node`, `{bSelectInside}`, `{bJumpToNode}`

- **node** 인수는 문서의 텍스트, 주석 또는 요소 노드입니다.
- 선택적인 **bSelectInside** 인수는 노드의 `innerHTML`을 선택할지 여부를 나타내는 부울 값입니다. 이 인수는 **node**가 요소 노드인 경우에만 관련이 있으며 생략되는 경우 기본값은 `false`입니다.
- 선택적인 **bJumpToNode** 인수는 필요한 경우 선택 영역을 표시하기 위해 문서 윈도우를 스크롤할지 여부를 나타내는 부울 값입니다. 이 인수가 생략되는 경우 기본값은 `false`입니다.

반환값

없음

dom.setSelection()

지원 버전

Dreamweaver 3

설명

문서의 선택 영역을 설정합니다.

인수

offsetBegin, offsetEnd

- 이러한 인수는 각각 새 선택 영역의 시작점 및 끝점이며, 문서 소스 코드까지의 문자 오프셋으로 표시됩니다. 두 숫자가 동일하면 새 선택 영역은 삽입점입니다. 새 선택 영역이 유효한 HTML 선택 영역이 아닌 경우 확장되어 문자를 첫 번째 유효한 HTML 선택 영역에 포함합니다. 예를 들어, **offsetBegin**과 **offsetEnd**가 `<IMG SRC="myImage.gif">` 내의 `SRC="myImage.gif"` 범위를 정의하는 경우 선택 영역이 확장되어 IMG 태그 전체를 포함합니다.

반환값

없음

dreamweaver.nodeExists()

지원 버전

Dreamweaver 3

설명

지정된 노드에 대한 참조가 여전히 유효한지 여부를 확인합니다. **Extension**을 작성하는 경우 노드를 참조한 다음 노드를 삭제하는 작업을 수행하는 경우가 많습니다(예: 노드 부모의 **innerHTML** 또는 **outerHTML** 속성 설정). 이 함수를 사용하면 노드의 속성이나 메서드를 참조하기 전에 노드가 삭제되지 않았는지 확인할 수 있습니다. 참조된 노드는 현재 문서에 없어도 됩니다.

인수

node

- node** 인수는 확인할 노드입니다.

반환값

부울 값을 반환합니다. 노드가 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

다음 예제에서는 현재 노드를 가져오고 그 안에서 표를 찾은 다음 나중에 **dw.nodeExists()**를 호출하여 원래 노드가 존재하는지 여부를 확인합니다.

```
function applyFormatToSelectedTable(){

    // get current selection
    var selObj = dw.getDocumentDOM().getSelectedNode();
    alternateRows(dwscripts.findDOMObject("presetNames").selectedIndex,
        findTable());

    // restore original selection, if it still exists; if not, just select the
    // table.
    var selArr;
    if (dw.nodeExists(selObj))
        selArr = dom.nodeToOffsets(selObj);
    else
        selArr = dom.nodeToOffsets(findTable());

    dom.setSelection(selArr[0],selArr[1]);
}
```

dreamweaver.selectAll()

지원 버전

Dreamweaver 3

설명

활성 문서 윈도우나 [사이트] 패널에서 [모두 선택] 작업을 수행합니다. Macintosh에서는 대화 상자나 부동 패널의 포커스가 있는 텍스트 필드에서 [모두 선택] 작업을 수행합니다.

참고: 활성 문서에서 작업이 수행되면 일반적으로 활성 문서의 모든 내용이 선택됩니다. 그러나 특정한 경우(예: 삽입점이 표 안에 있을 때)에는 활성 문서의 일부만 선택됩니다. 선택 영역을 문서 전체로 설정하려면 *dom.setSelection()* 함수를 사용합니다.

인수

없음

반환값

없음

활성자

448페이지의 “[dreamweaver.canSelectAll\(\)](#)”을 참조하십시오.

문자열 조작 함수

문자열 조작 함수를 사용하여 영어(Latin 1) 인코딩에서 플랫폼의 기본 인코딩으로 또는 그 반대로 문자열을 변환할 수 있을 뿐 아니라 문자열에 대한 정보를 가져올 수 있습니다.

dreamweaver.doURLEncoding()

지원 버전

Dreamweaver 1

설명

문자열을 받아 공백과 특수 문자를 모두 지정된 항목으로 대체하여 URL 인코딩된 문자열을 반환합니다.

인수

stringToConvert

- **stringToConvert** 인수는 함수에서 인코딩할, 인코딩되지 않은 URL을 포함하는 문자열입니다.

반환값

URL 인코딩된 문자열을 반환합니다.

예제

다음 예제에서는 "My URL-encoded string"의 URL.value를 보여 줍니다.

```
var URL = dw.doURLEncoding(theURL.value);  
returns "My%20URL-encoded%20string"
```

dreamweaver.getTokens()

지원 버전

Dreamweaver 1

설명

문자열을 받아 토큰으로 분할합니다.

인수

searchString, separatorCharacters

- **searchString** 인수는 토큰으로 분리할 문자열입니다.
- **separatorCharacters** 인수는 토큰의 마지막을 의미하는 문자입니다. 인용 문자열 내의 구분 문자는 무시됩니다. **separatorCharacters**에 포함된 모든 공백 문자(예: 탭)는 명시적으로 지정되지 않았어도 구분 문자로 간주됩니다. 공백 문자가 둘 이상 연속된 경우 단일 구분 기호로 간주됩니다.

반환값

토큰 문자열의 배열을 반환합니다.

예제

아래의 dw.getTokens() 함수 호출은 다음에 나오는 토큰을 반환합니다.

```
dreamweaver.getTokens('foo("my arg1", 34)', '(),')
```

- foo
- "my arg 1"
- 34

dreamweaver.latin1ToNative()

지원 버전

Dreamweaver 2

설명

영어(Latin 1) 인코딩의 문자열을 사용자 컴퓨터의 기본 인코딩으로 변환합니다. 이 함수는 확장 파일의 UI를 다른 언어로 표시하기 위한 것입니다.

참고: Windows 인코딩은 이미 영어(Latin 1)를 기반으로 하고 있으므로 Windows에서는 이 함수가 아무 영향을 미치지 않습니다.

인수

stringToConvert

- **stringToConvert** 인수는 영어(Latin 1) 인코딩에서 기본 인코딩으로 변환할 문자열입니다.

반환값

변환된 문자열을 반환합니다.

dreamweaver.nativeToLatin1()

지원 버전

Dreamweaver 2

설명

기본 인코딩의 문자열을 영어(Latin 1) 인코딩으로 변환합니다.

참고: Windows 인코딩은 이미 영어(Latin 1)를 기반으로 하고 있으므로 Windows에서는 이 함수가 아무 영향을 미치지 않습니다.

인수

stringToConvert

- **stringToConvert** 인수는 기본 인코딩에서 영어(Latin 1) 인코딩으로 변환할 문자열입니다.

반환값

변환된 문자열을 반환합니다.

dreamweaver.scanSourceString()

지원 버전

Dreamweaver UltraDev 1

설명

HTML의 문자열을 검색하여 태그, 속성, 지시문 및 텍스트를 찾습니다. `scanSourceString()` 함수는 발견하는 각 태그, 속성, 지시문 및 텍스트 영역에 대해 콜백 함수를 시작하며 이 콜백 함수는 사용자가 제공해야 합니다. **Dreamweaver**는 다음과 같은 콜백 함수를 지원합니다.

- `openTagBegin()`
- `openTagEnd()`
- `closeTagBegin()`
- `closeTagEnd()`
- `directive()`
- `attribute()`
- `text()`

Dreamweaver에서는 다음과 같은 경우에 7개의 콜백 함수를 호출합니다.

- 각각의 열기 태그(예: `</font>`)와 대응되는 닫기 태그(예: `<font>`) 및 빈 태그(예: `<img>` 또는 `<hr>`)에 대해 `openTagBegin()`이 호출됩니다. `openTagBegin()` 함수는 태그의 이름(예: "font" 또는 "img")과 문서에서 태그의 시작 부분 앞에 있는 바이트 수를 나타내는 문서 오프셋을 인수로 사용합니다. 이 함수는 검색을 계속해야 하면 `true`를 반환하고 중단해야 하면 `false`를 반환합니다.
- `openTagBegin()`이 실행된 후 각 HTML 속성에 대해 `attribute()`가 호출됩니다. `attribute()` 함수는 속성 이름을 포함하는 문자열(예: "color" 또는 "src")과 속성 값을 포함하는 문자열(예: "#000000" 또는 "foo.gif")을 인수로 사용합니다. `attribute()` 함수는 검색을 계속할지 여부를 나타내는 부울 값을 반환합니다.
- 태그의 모든 속성이 검색된 후 `openTagEnd()`가 호출됩니다. `openTagEnd()` 함수는 문서에서 열기 태그의 끝 부분 앞에 있는 바이트 수를 나타내는 문서 오프셋을 인수로 사용합니다. 이 함수는 검색을 계속할지 여부를 나타내는 부울 값을 반환합니다.

- 각각의 닫기 태그(예:)에 대해 `closeTagBegin()`이 호출됩니다. 이 함수는 닫기 태그의 이름(예: "font")과 문서에서 닫기 태그의 시작 부분 앞에 있는 바이트 수를 나타내는 문서 오프셋을 인수로 사용합니다. 이 함수는 검색을 계속할지 여부를 나타내는 부울 값을 반환합니다.
- `closeTagBegin()`이 반환된 후 `closeTagEnd()` 함수가 호출됩니다. `closeTagEnd()` 함수는 문서에서 닫기 태그의 끝 부분 앞에 있는 바이트 수를 나타내는 문서 오프셋을 인수로 사용합니다. 이 함수는 검색을 계속할지 여부를 나타내는 부울 값을 반환합니다.
- 각각의 HTML 주석, ASP 스크립트, JSP 스크립트 또는 PHP 스크립트에 대해 `directive()` 함수가 호출됩니다. `directive()` 함수는 지시문을 포함하는 문자열과 문서에서 닫기 태그의 끝 부분 앞에 있는 바이트 수를 나타내는 문서 오프셋을 인수로 사용합니다. 이 함수는 검색을 계속할지 여부를 나타내는 부울 값을 반환합니다.
- 문서에서 텍스트의 각 영역, 즉 태그나 지시문이 아닌 모든 것에 대해 `text()` 함수가 호출됩니다. 텍스트 영역에는 <title> 또는 <option> 태그 내의 텍스트처럼 사용자에게 보이지 않는 텍스트도 포함됩니다. `text()` 함수는 텍스트를 포함하는 문자열과 문서에서 닫기 태그의 끝 부분 앞에 있는 바이트 수를 나타내는 문서 오프셋을 인수로 사용합니다. `text()` 함수는 검색을 계속할지 여부를 나타내는 부울 값을 반환합니다.

인수

HTMLstr, parserCallbackObj

- **HTMLstr** 인수는 코드를 포함하는 문자열입니다.
- **parserCallbackObj** 인수는 `openTagBegin()`, `openTagEnd()`, `closeTagBegin()`, `closeTagEnd()`, `directive()`, `attribute()` 및 `text()` 메서드 중 하나 이상이 있는 JavaScript 객체입니다. 성능을 최상의 상태로 유지하려면 **parserCallbackObj**는 C 레벨 확장성 인터페이스를 사용하여 정의된 공유 라이브러리여야 합니다. **parserCallbackObj**가 필요한 콜백 함수만 정의하는 경우에도 성능이 향상됩니다.

반환값

부울 값을 반환합니다. 작업이 성공적으로 완료되면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

예제

다음 단계는 `dreamweaver.scanSourceString()` 함수를 사용하는 방법에 대한 예제입니다.

- 1 7개의 콜백 함수 중 하나 이상을 구현합니다.
- 2 `dreamweaver.scanSourceString()` 함수를 호출하는 스크립트를 작성합니다.
- 3 `dreamweaver.scanSourceString()` 함수가 HTML 및 포인터를 포함하는 문자열을 사용자가 작성한 콜백 함수에 전달합니다. 예를 들어, HTML 문자열이 "hello"라고 가정합니다.
- 4 문자열이 분석되고 문자열이 font 태그를 포함하는지 확인됩니다. 콜백 함수가 다음 순서로 호출됩니다.
 - `openTagBegin()` 함수
 - `attribute()` 함수(size 속성에 대해)
 - `openTagEnd()` 함수
 - `text()` 함수("hello" 문자열에 대해)
 - `closeTagBegin()` 및 `closeTagEnd()` 함수

변환 함수

변환 함수는 변환기를 직접 조작하거나 변환 결과를 조작합니다. 이러한 함수는 변환기를 실행하거나 변환기 정보를 가져오고, 잠금 영역의 내용을 편집하고, 선택 영역 오프셋을 가져오거나 설정할 때 사용할 변환된 소스를 지정합니다.

dom.runTranslator()

지원 버전

Dreamweaver 3

설명

이 함수는 문서에서 지정한 변환기를 실행합니다. 이 함수는 활성 문서에만 사용할 수 있습니다.

인수

translatorName

- **translatorName** 인수는 [변환] 환경 설정에 나타나는 변환기의 이름입니다.

반환값

없음

dreamweaver.editLockedRegions()

지원 버전

Dreamweaver 2

설명

이 함수는 인수의 값에 따라 잠긴 영역을 편집 가능 또는 편집 불가능으로 설정합니다. 기본적으로 잠긴 영역은 편집할 수 없습니다. 잠긴 영역을 이 함수를 통해 편집 가능으로 지정하지 않은 상태에서 편집하려고 하면 경고음이 울리고 변경 내용이 적용되지 않습니다.

참고: 잠긴 영역을 편집하면 라이브러리 항목 및 템플릿에 의도하지 않은 결과가 발생할 수 있습니다. 따라서 데이터 변환기의 컨텍스트 외부에서 이 함수를 사용하면 안 됩니다.

인수

bAllowEdits

- **bAllowEdits** 인수는 부울 값입니다. **true**인 경우 편집이 허용되고, **false**인 경우 편집이 허용되지 않습니다. 이 함수를 호출하는 스크립트의 실행이 끝나면 자동으로 잠긴 영역이 기본 상태(편집 불가능)로 복원됩니다.

반환값

없음

dreamweaver.getTranslatorList()

지원 버전

Dreamweaver 3

설명

이 함수는 설치된 변환기의 목록을 가져옵니다.

인수

없음

반환값

문자열 배열을 반환합니다. 여기서 각 문자열은 [변환] 환경 설정에 표시되는 변환기의 이름을 나타냅니다.

dreamweaver.useTranslatedSource()**지원 버전**

Dreamweaver 2

설명

이 함수는 dom.nodeToOffsets() 및 dom.getSelection()이 반환하는 값을 지정합니다. 이러한 값은 dom.offsetsToNode() 및 dom.setSelection()에서 사용되며, 변환되지 않은 소스가 아니라 변환된 소스(변환기 실행 후 DOM에 들어 있는 HTML)까지의 오프셋이어야 합니다.

참고: 이 함수는 속성 관리자 파일에서만 유효합니다.

인수**bUseTranslatedSource**

- **bUseTranslatedSource** 인수는 부울 값입니다. true인 경우 함수가 변환된 소스까지의 오프셋을 사용하고, false인 경우 변환되지 않은 소스를 사용합니다.

인수의 기본값은 false입니다. dw.useTranslatedSource()를 호출하는 스크립트의 실행이 끝났을 때 이전에 dw.useTranslatedSource()를 false 인수로 명시적으로 호출하지 않은 경우 자동으로 이후의 dw.getSelection(), dw.setSelection(), dw.nodeToOffsets() 및 dw.offsetsToNode() 호출에 변환되지 않은 소스가 사용됩니다.

반환값

없음

XSLT 함수

XSLT 함수는 XML 파일을 처리합니다. 이러한 함수는 스키마 트리 또는 XML 문서에 대한 참조 등의 XML 문서 정보를 가져오며, 현재 XSLT 문서와 연관된 XML 문서를 지정하라는 대화 상자를 사용자에게 표시합니다.

MMXSLT.getXML()**지원 버전**

Dreamweaver CS3

설명

XML 파일에 사용할 XML 소스 문자열을 가져옵니다.

인수**xmlSourceURI**

- XML 파일에 대한 URI를 나타내는 문자열로, 절대적(http 또는 https)이거나, 사이트에 상대적 또는 문서에 상대적일 수 있습니다.

반환값

XML 파일의 내용을 포함하는 문자열을 반환합니다.

예제

```
var xmlSource = MMXSLT.getXML(this.fileDataSetURL);
```

MMXSLT.getXMLSchema()

지원 버전

Dreamweaver 8

설명

이 함수는 지정한 XML 파일에 대한 스키마 트리를 반환합니다.

인수

schemaURI, {bRefresh}

- 필수적인 **schemaURI** 인수는 로컬 또는 원격 XML 파일에 대한 참조를 나타내는 문자열입니다.
- 선택적인 **bRefresh** 인수는 부울 값입니다. **true**인 경우 스키마가 새로 고쳐지고, **false**인 경우 XML 스키마 캐시에서 스키마의 사본이 반환됩니다. 기본값은 **false**입니다.

반환값

XML 스키마 트리를 포함하는 문자열을 반환합니다.

예제

다음 예제에서는 **menus.xml**에 대해 XML 스키마 캐시에서 스키마 트리를 가져옵니다.

```
var theSchema = MMXSLT.getXMLSchema("file:///c:/Program Files/Adobe/~/
  Adobe Dreamweaver CS4/Configuration/Menus/menus.xml");
```

MMXSLT.getXMLSourceURI()

지원 버전

Dreamweaver 8

설명

이 함수는 현재 XSLT 문서와 연관된 XML 소스 문서에 대한 참조를 가져옵니다.

인수

xsltfileURI, {bUseTempForRemote}

- xsltfileURI** 인수는 XSL 파일의 위치를 가리키는 로컬 파일 URI 문자열입니다.
- 선택적인 **bUseTempForRemote** 인수는 부울 값입니다. **true**인 경우 원본 XML 파일이 원격일 때(예: <http://myHost/rssfeed.xml>) 다운로드되는 임시 XML 파일에 대한 참조(예: [file:///C:/Documents and Settings/username/Local Settings/Temporary Internet Files/Content.IE5/GTSLQ9KZ/rss\[1\].xml](file:///C:/Documents and Settings/username/Local Settings/Temporary Internet Files/Content.IE5/GTSLQ9KZ/rss[1].xml))가 반환되고, **false**인 경우 절대 참조가 반환됩니다.

반환값

현재 XSLT 문서와 연관된 XML 소스 문서에 대한 참조를 포함하는 문자열을 반환합니다. XML 소스 참조가 원격 참조인 경우 이 함수는 다운로드되는 파일 경로를 임시 위치로 반환합니다.

예제

다음 예제에서는 c:\myxslt\myxsltdocument.xml과 연관된 XML 소스 문서에 대한 참조를 가져옵니다.

```
var theXMLSource = MMXSLT.getXMLSourceURI("file:///c:/myxslt/myxsltdocument.xml");
```

MMXSLT.launchXMLSourceDialog()**지원 버전**

Dreamweaver 8

설명

이 함수는 현재 XSLT 문서와 연관된 XML 소스 문서를 지정하라는 대화 상자를 표시합니다. 사용자는 XML 문서에 대한 로컬 또는 원격 참조를 선택할 수 있습니다.

인수

{xsltfileURI}, {bUseTempForRemote}, {bAddSchemaReference}

- 선택적인 **xsltfileURI** 인수는 XSL 파일의 위치를 가리키는 로컬 파일 URI 문자열입니다. 이 인수가 생략되는 경우 기본값은 현재 열려 있는 문서입니다.
- 선택적인 **bUseTempForRemote** 인수는 부울 값입니다. true인 경우 원본 XML 파일이 원격일 때(예: http://myHost/rssfeed.xml) 다운로드되는 임시 XML 파일에 대한 참조(예: file:///C:/Documents and Settings/username/Local Settings/Temporary Internet Files/Content.IE5/GTSLQ9KZ/rss[1].xml)가 반환되고, false인 경우 절대 참조가 반환됩니다.
- 선택적인 **bAddSchemaReference** 인수는 XML 소스 대화 상자에 지정된 XML 소스 URI를 가리키는 현재 문서에 참조를 추가합니다. 이 인수가 생략되는 경우 기본값은 현재 열려 있는 문서입니다.

반환값

현재 XSLT 문서와 연관된 XML 소스 문서에 대한 참조를 포함하는 문자열을 반환합니다. XML 소스 참조가 원격 참조인 경우 이 함수는 다운로드되는 파일 경로를 임시 위치로 반환합니다.

예제

다음 예제에서는 값을 지정하지 않고 [XML 소스 문서] 대화 상자를 실행합니다.

```
MMXSLT.launchXMLSourceDialog()
```

15장: 페이지 내용

Adobe® Dreamweaver® 페이지 내용 함수는 웹 페이지의 내용에 영향을 미치는 작업을 수행합니다. 이러한 작업은 다음과 같습니다.

- 에셋 패널에서 에셋 조작
- 비헤이비어 추가
- 클립보드에서 요소 잘라내기 및 붙여넣기
- 템플릿 적용
- 코드 단편 삽입
- Spry XML 데이터 세트 만들기
- Spry 및 기타 위젯 편집 기능 향상
- 위젯 삽입
- 브라우저 호환성 확인 함수를 사용하여 다양한 브라우저에서 작동하는 페이지 레이아웃 생성

에셋 패널 함수

에셋 패널로 API에 프로그래밍된 에셋 패널 함수를 사용하면 [에셋] 패널의 요소(템플릿, 라이브러리, 이미지, Adobe Shockwave 및 Adobe Flash 내용, URL, 색상 및 스크립트)를 관리하고 사용할 수 있습니다.

dreamweaver.assetPalette.addToFavoritesFromDocument()

지원 버전

Dreamweaver 4

설명

[문서] 윈도우에서 선택한 요소를 [즐겨찾기] 목록에 추가합니다. 이 함수는 이미지, Shockwave 파일, Flash 파일, 텍스트 글꼴 색상 및 URL만 처리합니다.

인수

없음

반환값

없음

dreamweaver.assetPalette.addToFavoritesFromSiteAssets()

지원 버전

Dreamweaver 4

설명

[사이트] 목록에서 선택한 요소를 [즐거찾기] 목록에 추가하고 [즐거찾기] 목록에서 각 항목에 별칭을 지정합니다. 그러나 이 함수는 [사이트] 목록에서 요소를 제거하지는 않습니다.

인수

없음

반환값

없음

dreamweaver.assetPalette.addToFavoritesFromSiteWindow()

지원 버전

Dreamweaver 4

설명

[사이트] 패널에서 선택한 요소를 [즐거찾기] 목록에 추가합니다. 이 함수는 이미지, 동영상, 스크립트, Shockwave 파일 및 FLA 파일만 처리합니다. 선택한 다른 폴더나 파일은 무시됩니다.

인수

없음

반환값

없음

dreamweaver.assetPalette.copyToSite()

지원 버전

Dreamweaver 4

설명

선택한 요소를 다른 사이트에 복사하고 해당 사이트의 [즐거찾기] 목록에 넣습니다. 선택한 요소가 색상이나 URL이 아닌 파일이면 실제 파일이 해당 사이트에 복사됩니다.

인수

targetSite

- **targetSite** 인수는 site.getSites() 호출에서 반환된 대상 사이트의 이름입니다.

반환값

없음

dreamweaver.assetPalette.edit()

지원 버전

Dreamweaver 4

설명

선택한 요소를 기본 외부 편집기나 [사용자 정의 편집] 컨트롤로 편집합니다. 색상의 경우에는 색상 선택기가 나타납니다. URL의 경우에는 URL 및 별칭을 입력하도록 요구하는 대화 상자가 나타납니다. 색상 및 URL의 [사이트] 목록에는 이 함수를 사용할 수 없습니다.

인수

없음

반환값

없음

활성자

441 페이지의 “[dreamweaver.assetPalette.canEdit\(\)](#)”를 참조하십시오.

dreamweaver.assetPalette.getSelectedCategory()

지원 버전

Dreamweaver 4

설명

현재 선택한 범주를 반환합니다.

인수

없음

반환값

현재 선택한 범주를 반환합니다. 이 범주는 "templates", "library", "images", "movies", "shockwave", "flash", "scripts", "colors" 또는 "urls" 중 하나일 수 있습니다.

dreamweaver.assetPalette.getSelectedItems()

지원 버전

Dreamweaver 4

설명

[에셋] 패널의 [사이트] 목록이나 [즐거찾기] 목록에서 선택한 항목의 배열을 반환합니다.

인수

없음

반환값

선택한 각 항목에 대해 다음 세 문자열의 배열을 반환합니다.

- **name** 문자열 - [에셋] 패널에 나타나는 이름/파일 이름 또는 별칭
- **value** 문자열 - 전체 경로, 전체 URL 또는 색상 값(선택한 항목에 따라 다름)

- **type** 문자열 - "folder" 또는 "templates", "library", "images", "movies", "shockwave", "flash", "scripts", "colors" 또는 "urls" 범주 중 하나

참고: [에셋] 패널에서 아무 것도 선택하지 않은 경우 빈 문자열 하나로 구성된 배열이 반환됩니다.

예제

URL이 범주인 경우 [즐거찾기] 목록에서 MyFolderName이라는 폴더와 MyFavoriteURL이라는 URL을 모두 선택하면 이 함수는 다음 목록을 반환합니다.

```
items[0] = "MyFolderName"  
items[1] = "//path/FolderName"  
items[2] = "folder"  
items[3] = "MyFavoriteURL"  
items[4] = "http://www.MyFavoriteURL.com"  
items[5] = "urls"
```

dreamweaver.assetPalette.getSelectedView()

지원 버전

Dreamweaver 4

설명

현재 [에셋] 패널에 표시되는 목록을 나타냅니다.

인수

없음

반환값

값이 "site" 또는 "favorites"인 문자열을 반환합니다.

dreamweaver.assetPalette.insertOrApply()

지원 버전

Dreamweaver 4

설명

선택한 요소를 삽입하거나 현재 선택 영역에 요소를 적용합니다. 선택 영역에 템플릿, 색상 및 URL을 적용하고 삽입점에 URL 및 기타 요소를 삽입합니다. 문서가 열려 있지 않으면 이 함수를 사용할 수 없습니다.

인수

없음

반환값

없음

활성자

442페이지의 “[dreamweaver.assetPalette.canInsertOrApply\(\)](#)”를 참조하십시오.

dreamweaver.assetPalette.locateInSite()

지원 버전

Dreamweaver 4

설명

[사이트] 패널의 로컬측에서 선택한 요소와 연관된 파일을 선택합니다. 색상이나 URL에는 이 함수를 사용할 수 없지만 [사이트] 목록과 [즐거찾기] 목록에서는 사용할 수 있습니다. [즐거찾기] 목록에서 선택한 폴더는 무시됩니다.

인수

없음

반환값

없음

dreamweaver.assetPalette.newAsset()

지원 버전

Dreamweaver 4

설명

[즐거찾기] 목록의 현재 범주에 새 요소를 만듭니다. 라이브러리와 템플릿의 경우에는 사용자가 즉시 이름을 지정할 수 있는 새로운 빈 라이브러리나 템플릿 파일이 만들어지고, 색상의 경우에는 색상 선택기가 나타납니다. URL의 경우에는 URL 및 별칭을 입력하도록 요구하는 대화 상자가 나타납니다. 이미지, Shockwave 파일, Flash 파일 또는 스크립트에는 이 함수를 사용할 수 없습니다.

인수

없음

반환값

없음

dreamweaver.assetPalette.newFolder()

지원 버전

Dreamweaver 4

설명

기본 이름(무제)으로 현재 범주에 새 폴더를 만들고 기본 이름 주위에 텍스트 상자를 넣습니다. [즐거찾기] 목록에만 사용할 수 있습니다.

인수

없음

반환값

없음

dreamweaver.assetPalette.recreateLibraryFrom Document()

지원 버전

Dreamweaver 4

설명

사용하지 않는 libraryPalette 함수 recreateLibraryFromDocument()를 대체합니다. 현재 문서에서 라이브러리 항목의 선택한 인스턴스에 대한 LBI(라이브러리 항목) 파일을 만듭니다. 이 함수는 속성 관리자에서 [다시 생성]을 클릭하는 것과 같습니다.

인수

없음

반환값

없음

dreamweaver.assetPalette.refreshSiteAssets()

지원 버전

Dreamweaver 4

설명

사이트를 검색하고 [사이트] 목록으로 전환하여 목록을 채웁니다.

인수

없음

반환값

없음

dreamweaver.assetPalette.removeFromFavorites()

지원 버전

Dreamweaver 4

설명

[즐거찾기] 목록에서 선택한 요소를 제거합니다. 이 함수를 사용하여 디스크의 실제 파일을 삭제할 수는 없습니다. 단, 파일을 삭제하기 전에 메시지를 표시하는 라이브러리나 템플릿의 경우는 예외입니다. 이 함수는 범주가 라이브러리나 템플릿인 경우 또는 [즐거찾기] 목록에서만 적용됩니다.

인수

없음

반환값

없음

dreamweaver.assetPalette.renameNickname()

지원 버전

Dreamweaver 4

설명

기존 별칭 주위에 텍스트 상자를 표시하여 폴더 이름 또는 파일 별칭을 편집합니다. 라이브러리 또는 템플릿 범주나 [즐거찾기] 목록에서만 사용할 수 있습니다.

인수

없음

반환값

없음

dreamweaver.assetPalette.setSelectedCategory()

지원 버전

Dreamweaver 4

설명

전환하여 다른 범주를 나타냅니다.

인수

categoryType

- **categoryType** 인수는 "templates", "library", "images", "movies", "shockwave", "flash", "scripts", "colors" 또는 "urls" 범주 중 하나일 수 있습니다.

반환값

없음

dreamweaver.assetPalette.setSelectedView()

지원 버전

Dreamweaver 4

설명

표시를 전환하여 [사이트] 목록이나 [즐거찾기] 목록을 표시합니다.

인수

viewType

- **viewType** 인수는 "site" 또는 "favorites"인 문자열입니다.

반환값

없음

dreamweaver.referencePalette.getFontSize()

지원 버전

Dreamweaver 4

설명

[참조] 패널 표시 영역의 현재 글꼴 크기를 반환합니다.

인수

없음

반환값

small, medium 또는 large와 같은 상대적 글꼴 크기를 반환합니다.

dreamweaver.referencePalette.setFontSize()

지원 버전

Dreamweaver 4

설명

[참조] 패널에 나타나는 글꼴 크기를 변경합니다.

인수

fontSize

- **fontSize** 인수는 상대적 크기를 나타내는 small, medium 또는 large 중 하나입니다.

반환값

없음

비헤이비어 함수

비헤이비어 함수를 사용하면 객체에 비헤이비어를 추가하거나 객체에서 비헤이비어를 제거하고, 객체에 첨부된 비헤이비어를 찾고, 비헤이비어가 첨부된 객체에 대한 정보를 가져오는 등의 작업을 수행할 수 있습니다. `dreamweaver.behaviorInspector` 객체의 메서드는 현재 문서의 선택 영역이 아니라 [비헤이비어] 패널의 선택 영역을 제어하거나 선택 영역에 대해 작용합니다.

dom.addBehavior()

지원 버전

Dreamweaver 3

설명

선택한 요소에 새 이벤트/액션 쌍을 추가합니다. 이 함수는 활성 문서에만 사용할 수 있습니다.

인수

event, action, {eventBasedIndex}

- **event** 인수는 비헤이비어를 요소에 첨부할 JavaScript 이벤트 핸들러입니다(예: `onClick`, `onMouseOver` 또는 `onLoad`).
- **action** 인수는 [비헤이비어] 패널을 사용하여 액션이 추가되는 경우 `applyBehavior()`가 반환하는 함수 호출입니다(예: `"MM_popupMsg('Hello World')"`).
- 선택적인 **eventBasedIndex** 인수는 이 액션이 추가될 위치입니다. **eventBasedIndex** 인수는 0부터 시작하는 인덱스입니다. 두 액션이 지정된 이벤트와 이미 연관되어 있는 경우 **eventBasedIndex**를 1로 지정하면 이 액션은 다른 두 액션 사이에 실행됩니다. 이 인수를 생략하면 액션은 지정된 이벤트의 모든 기존 액션 뒤에 추가됩니다.

반환값

없음

dom.getBehavior()

지원 버전

Dreamweaver 3

설명

지정된 이벤트 내의 지정된 위치에서 액션을 가져옵니다. 이 함수는 현재 선택 영역에 대해 작용하고 활성 문서에서만 유효합니다.

인수

event, {eventBasedIndex}

- **event** 인수는 액션을 요소에 첨부하는 데 사용되는 JavaScript 이벤트 핸들러입니다(예: `onClick`, `onMouseOver` 또는 `onLoad`).
- 선택적인 **eventBasedIndex** 인수는 가져올 액션의 위치입니다. 예를 들어, 두 액션이 지정된 이벤트와 연관되어 있으면 0이 첫 번째이고 1이 두 번째입니다. 이 인수를 생략하면 지정된 이벤트의 모든 액션이 반환됩니다.

반환값

함수 호출을 나타내는 문자열(예: `"MM_swapImage('document.Image1','document.Image1','foo.gif','#933292969950')"`)을 반환하거나, **eventBasedIndex**가 생략된 경우에는 문자열 배열을 반환합니다.

dom.reapplyBehaviors()

지원 버전

Dreamweaver 3

설명

지정된 노드의 비헤이비어 호출과 연관된 함수가 문서의 HEAD 섹션에 있는지 확인하고 없으면 삽입합니다.

인수

elementNode

- **elementNode** 인수는 현재 문서 내의 요소 노드입니다. 이 인수를 생략하면 문서의 모든 요소 노드에서 고립된 비헤이비어 호출이 있는지 여부가 확인됩니다.

반환값
없음

dom.removeBehavior()

지원 버전
Dreamweaver 3

설명
지정된 이벤트 내의 지정된 위치에서 액션을 제거합니다. 이 함수는 현재 선택 영역에 대해 작용하고 활성 문서에서만 유효합니다.

인수
event, {**eventBasedIndex**}

- **event** 인수는 액션을 요소에 첨부하는 데 사용되는 이벤트 핸들러입니다(예: **onClick**, **onMouseOver** 또는 **onLoad**). 이 인수를 생략하면 요소에서 모든 액션이 제거됩니다.
- 선택적인 **eventBasedIndex** 인수는 제거할 액션의 위치입니다. 예를 들어, 두 액션이 지정된 이벤트와 연관되어 있으면 0이 첫 번째이고 1이 두 번째입니다. 이 인수를 생략하면 지정된 이벤트의 모든 액션이 제거됩니다.

반환값
없음

dreamweaver.getBehaviorElement()

지원 버전
Dreamweaver 2, CS4에서 업데이트됨

설명
비헤이비어가 적용되는 태그에 해당하는 DOM 객체를 가져옵니다. 이 함수는 비헤이비어 액션 파일에만 적용할 수 있습니다.

인수
없음

반환값
DOM 객체 또는 null 값을 반환합니다. 이 함수는 다음과 같은 상황에서 null 값을 반환합니다.

- 현재 스크립트가 [비헤이비어] 패널의 컨텍스트 내에서 실행되지 않는 경우
- **dreamweaver.popupAction()**에서 현재 실행 중인 스크립트를 시작하는 경우
- [비헤이비어] 패널이 링크 래퍼에 이벤트를 첨부하려고 하지만 링크 래퍼가 없는 경우
- 이 함수가 액션 파일 외부에 나타나는 경우

예제

278페이지의 “[dreamweaver.getBehaviorTag\(\)](#)”와 같은 방식으로 `dreamweaver.getBehaviorElement()` 함수를 사용하여 선택한 액션이 선택한 HTML 태그에 적합한지 여부를 확인할 수 있습니다. 차이점은 태그와 해당 속성에 대한 더 자세한 정보에 액세스할 수 있다는 것입니다. 또 다른 프레임이나 윈도우를 대상으로 하지 않는 하이퍼텍스트 링크(A HREF)에만 적용할 수 있는 액션을 작성하는 경우 `getBehaviorElement()` 함수를 사용할 수 있습니다. [매개 변수] 대화 상자의 사용자 인터페이스를 초기화하는 함수의 일부로 `getBehaviorElement()` 함수를 사용할 수 있습니다. 예를 들면 다음과 같습니다.

```
function initializeUI(){
    var theTag = dreamweaver.getBehaviorElement();
    var CANBEAPPLIED = (theTag.tagName == "A" && ~
theTag.getAttribute("HREF") != null && ~
theTag.getAttribute("TARGET") == null);
    if (CANBEAPPLIED) {
        // display the action user interface
    } else{
        // display a helpful message that tells the user
        // that this action can only be applied to a
        // link without an explicit target]
    }
}
```

`dreamweaver.getBehaviorTag()`

지원 버전

Dreamweaver 1.2

설명

비헤이비어가 적용되는 태그의 소스를 가져옵니다. 이 함수는 액션 파일에만 적용할 수 있습니다.

인수

없음

반환값

태그의 소스를 나타내는 문자열을 반환합니다. 이 문자열은 `canAcceptBehavior()` 함수에 인수(`HTMLelement`)로 전달되는 문자열과 같은 문자열입니다. 이 함수가 액션 파일 외부에 나타나는 경우 반환값은 빈 문자열입니다.

예제

하이퍼텍스트 링크(A HREF)에만 적용할 수 있는 액션을 작성하는 경우 다음 예제와 같이 [매개 변수] 대화 상자의 사용자 인터페이스를 초기화하는 함수로 `getBehaviorTag()` 함수를 사용할 수 있습니다.

```
function initializeUI(){
    var theTag = dreamweaver.getBehaviorTag().toUpperCase();
    var CANBEAPPLIED = (theTag.indexOf('HREF') != -1);
    if (CANBEAPPLIED) {
        // display the action UI
    } else{
        // display a helpful message that tells the user
        // that this action can only be applied to a
        // hyperlink
    }
}
```

dreamweaver.popupAction()

지원 버전

Dreamweaver 2, CS4에서 업데이트됨

설명

지정된 비헤이비어 액션에 대해 [매개 변수] 대화 상자를 시작합니다. 사용자 입장에서 [비헤이비어] 패널의 [액션] 팝업 메뉴에서 액션을 선택하는 것과 같습니다. 이 함수를 사용하면 액션이 아닌 **Extension** 파일에서 사용자 문서의 객체에 비헤이비어를 추가할 수 있습니다. 또한 이 함수는 사용자가 대화 상자를 종료할 때까지 다른 편집 작업을 차단합니다.

참고: 이 함수는 *objectTag()* 함수 내부, 명령 파일의 스크립트 또는 속성 관리자 파일에서 호출할 수 있습니다.

인수

actionName, {funcCall}

- actionName 인수는 Configuration/Behaviors/Actions 폴더의 파일 이름을 포함하는 문자열입니다. 이 파일에는 JavaScript 비헤이비어 액션이 포함됩니다(예: "Swap Image.htm").
- 선택적인 funcCall 인수는 actionName에 지정된 액션에 대한 함수 호출을 포함하는 문자열입니다(예: "MM_SwapImage(...)"). 액션 파일의 applyBehavior() 함수는 지정된 경우 이 인수를 제공합니다.

반환값

비헤이비어 액션에 대한 함수 호출을 반환합니다. 사용자가 [매개 변수] 대화 상자에서 [확인]을 클릭하면 비헤이비어가 현재 문서에 추가됩니다. 해당 함수는 문서의 HEAD 섹션에 추가되고, HTML은 BODY 섹션의 맨 위에 추가되며, 문서에서 다른 편집 작업을 수행할 수도 있습니다. 함수 호출(예: "MM_SwapImage(...)")은 문서에 추가되지는 않지만 이 함수의 반환값이 됩니다.

dreamweaver.behaviorInspector.getBehaviorAt()

지원 버전

Dreamweaver 3

설명

[비헤이비어] 패널의 지정된 위치에서 이벤트/액션 쌍을 가져옵니다.

인수

positionIndex

- positionIndex 인수는 [비헤이비어] 패널에서 액션의 위치입니다. 목록에서 첫 번째 액션의 위치는 0입니다.

반환값

다음 두 항목의 배열을 반환합니다.

- 이벤트 핸들러
- 함수 호출 또는 JavaScript 명령문

예제

positionIndex는 0부터 시작하는 인덱스이므로, [비헤이비어] 패널에 목록이 표시되는 경우

dreamweaver.behaviorInspector.getBehaviorAt(2) 함수를 호출하면 "onMouseOver" 및

"MM_changeProp('document.moon','document.moon','src','sun.gif','MG')" 문자열이 포함된 배열이 반환됩니다.

dreamweaver.behaviorInspector.getBehaviorCount()

지원 버전

Dreamweaver 3

설명

이벤트 핸들러를 통해 현재 선택한 요소에 첨부된 액션 수를 셉니다.

인수

없음

반환값

요소에 첨부된 액션 수를 나타내는 정수를 반환합니다. 이 숫자는 [비헤이비어] 패널에 표시되는 액션 수와 같고 Dreamweaver 비헤이비어 액션과 사용자 정의 JavaScript를 포함합니다.

예제

선택한 링크 에 대해 dreamweaver.behaviorInspector.getBehaviorCount()를 호출하면 2가 반환됩니다.

dreamweaver.behaviorInspector.getSelectedBehavior()

지원 버전

Dreamweaver 3

설명

[비헤이비어] 패널에서 선택한 액션의 위치를 가져옵니다.

인수

없음

반환값

[비헤이비어] 패널에서 선택한 액션의 위치를 나타내는 정수를 반환하거나, 선택된 액션이 없는 경우에는 -1을 반환합니다.

예제

[비헤이비어] 패널의 첫 번째 액션이 선택된 경우 dreamweaver.behaviorInspector.getSelectedBehavior() 함수를 호출하면 0이 반환됩니다.

dreamweaver.behaviorInspector.moveBehaviorDown()

지원 버전

Dreamweaver 3

설명

이벤트의 범위 내에서 실행 순서를 변경하여 비헤이비어 액션을 아래로 이동합니다.

인수

positionIndex

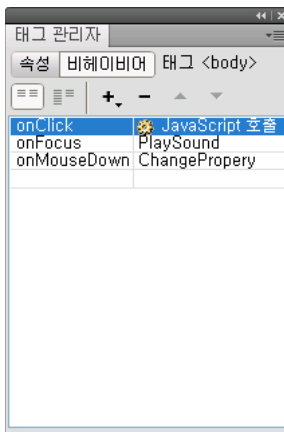
- **positionIndex** 인수는 [비헤이비어] 패널에서 액션의 위치입니다. 목록에서 첫 번째 액션의 위치는 0입니다.

반환값

없음

예제

`dreamweaver.behaviorInspector.moveBehaviorDown(2)` 함수를 호출하면 `onMouseDown` 이벤트에서 [이미지 미리 로드하기] 및 [속성 바꾸기] 액션의 위치가 바뀝니다. `onClick` 및 `onFocus` 이벤트에는 각각 하나의 비헤이비어만 연관되어 있으며 `onMouseDown` 이벤트 그룹의 아래쪽에 위치 3의 비헤이비어가 이미 있으므로 다른 위치에 대해 `dreamweaver.behaviorInspector.moveBehaviorDown()` 함수를 호출해도 아무 효과가 없습니다.



참조

280페이지의 “[dreamweaver.behaviorInspector.getSelectedBehavior\(\)](#)”

dreamweaver.behaviorInspector.moveBehaviorUp()

지원 버전

Dreamweaver 3

설명

이벤트의 범위 내에서 실행 순서를 변경하여 비헤이비어를 위로 이동합니다.

인수

positionIndex

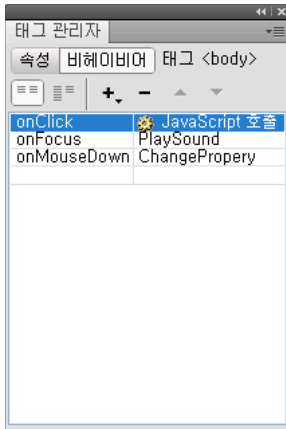
- **positionIndex** 인수는 [비헤이비어] 패널에서 액션의 위치입니다. 목록에서 첫 번째 액션의 위치는 0입니다.

반환값

없음

예제

`dreamweaver.behaviorInspector.moveBehaviorUp(3)` 함수를 호출하면 `onMouseOver` 이벤트에서 [이미지 미리 로드하기] 및 [속성 바꾸기] 액션의 위치가 바뀝니다. `onClick` 및 `onFocus` 이벤트에는 각각 하나의 비헤이비어만 연관되어 있으며 `onMouseDown` 이벤트 그룹의 위쪽에 위치 2의 비헤이비어가 이미 있으므로 다른 위치에 대해 `dreamweaver.behaviorInspector.moveBehaviorUp()` 함수를 호출해도 아무 효과가 없습니다.



참조

280페이지의 “[dreamweaver.behaviorInspector.getSelectedBehavior\(\)](#)”

dreamweaver.behaviorInspector.setSelectedBehavior()

지원 버전

Dreamweaver 3

설명

[비헤이비어] 패널의 지정된 위치에서 액션을 선택합니다.

인수

positionIndex

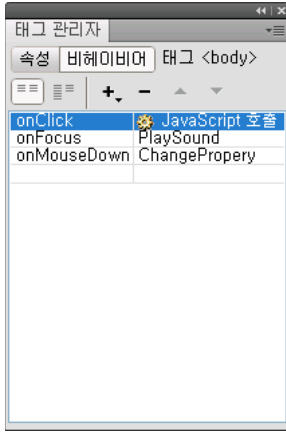
- **positionIndex** 인수는 [비헤이비어] 패널에서 액션의 위치입니다. 목록에서 첫 번째 액션의 위치는 0입니다. 모든 액션의 선택을 취소하려면 **positionIndex**를 -1로 지정합니다. 액션이 없는 위치를 지정하는 것은 -1을 지정하는 것과 같습니다.

반환값

없음

예제

dreamweaver.behaviorInspector.setSelection(2) 함수를 호출하면 onMouseDown 이벤트와 연관된 [속성 바꾸기] 액션이 선택됩니다.



참조

280페이지의 “[dreamweaver.behaviorInspector.getSelectedBehavior\(\)](#)”

클립보드 함수

클립보드 함수는 잘라내기, 복사 및 붙여넣기와 관련됩니다. Macintosh의 경우 일부 클립보드 함수는 대화 상자와 부동 패널의 텍스트 필드에도 적용할 수 있습니다. 텍스트 필드에 적용할 수 있는 함수는 dreamweaver 객체의 메서드와 DOM 객체의 메서드로 구현됩니다. dreamweaver 버전의 함수는 활성 윈도우, 즉 현재 [문서] 윈도우, 코드 관리자 또는 [사이트] 패널의 선택 영역에 적용됩니다. 또한 Macintosh에서 이 함수는 현재 필드에 있는 경우 텍스트 상자의 선택 영역에도 적용할 수 있습니다. DOM 버전의 함수는 지정된 문서의 선택 영역에 항상 적용됩니다.

dom.clipCopy()

지원 버전

Dreamweaver 3

설명

선택 영역을 정의하는 HTML 마크업과 함께 선택 영역을 클립보드에 복사합니다.

인수

없음

반환값

없음

dom.clipCopyText()

지원 버전

Dreamweaver 3

설명

선택한 텍스트를 클립보드에 복사하고, HTML 마크업은 포함하지 않습니다.

인수

없음

반환값

없음

활성자

432페이지의 “[dom.canClipCopyText\(\)](#)”를 참조하십시오.

dom.clipCut()

지원 버전

Dreamweaver 3

설명

선택 영역을 정의하는 HTML 마크업과 함께 선택 영역을 제거하여 클립보드에 놓습니다.

인수

없음

반환값

없음

dom.clipPaste()

지원 버전

Dreamweaver 3

설명

현재 문서에서 현재 삽입점이나 현재 선택 영역에 클립보드의 내용을 붙여 넣습니다. 클립보드에 HTML이 포함되어 있는 경우에는 그 자체로 해석됩니다.

인수

없음

반환값

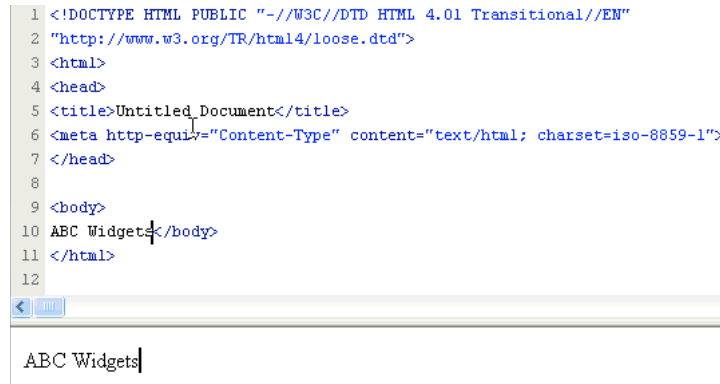
없음

활성자

432페이지의 “[dom.canClipPaste\(\)](#)”를 참조하십시오.

예제

클립보드에 ABC Widgets가 포함되어 있는 경우 dw.getDocumentDOM().clipPaste()를 호출하면 다음과 같은 결과가 나타납니다.



dreamweaver.clipCopy()

지원 버전

Dreamweaver 3

설명

활성 [문서] 윈도우, 대화 상자, 부동 패널 또는 [사이트] 패널에서 현재 선택 영역을 클립보드에 복사합니다.

인수

없음

반환값

없음

활성자

442페이지의 “[dreamweaver.canClipCopy\(\)](#)”를 참조하십시오.

dreamweaver.clipCut()

지원 버전

Dreamweaver 3

설명

활성 [문서] 윈도우, 대화 상자, 부동 패널 또는 [사이트] 패널에서 현재 선택 영역을 제거하여 클립보드에 놓습니다.

인수

없음

반환값

없음

활성자

442페이지의 “[dreamweaver.canClipCut\(\)](#)”을 참조하십시오.

dreamweaver.clipPaste()

지원 버전

Dreamweaver 3 Dreamweaver 8에 **strPasteOption** 인수가 추가되었습니다.

설명

현재 문서, 대화 상자, 부동 패널 또는 [사이트] 패널에 클립보드의 내용을 붙여 넣습니다.

인수

{strPasteOption}

- 선택적인 **strPasteOption** 인수는 실행할 붙여넣기 작업의 유형을 지정합니다. 값에는 "text", "structured", "basicFormat" 및 "fullFormat"이 포함됩니다.

반환값

없음

활성자

443페이지의 “[dreamweaver.canClipPaste\(\)](#)”를 참조하십시오.

예제

다음 예제에서는 클립보드의 내용을 텍스트로 붙여 넣습니다.

```
dw.clipPaste("text");
```

dreamweaver.getClipboardText()

지원 버전

Dreamweaver 3

설명

클립보드에 저장된 모든 텍스트를 가져옵니다.

인수

{bAsText}

- 선택적인 **bAsText** 부울 값은 클립보드 내용을 텍스트로 가져올지 여부를 지정합니다. **bAsText**가 true이면 클립보드 내용을 텍스트로 가져옵니다. **bAsText**가 false이면 내용의 서식이 유지됩니다. 이 인수의 기본값은 false입니다.

반환값

클립보드에 텍스트(HTML일 수 있음)가 포함되어 있으면 클립보드의 내용을 포함하는 문자열을 반환하고, 그렇지 않으면 아무 것도 반환하지 않습니다.

예제

`dreamweaver.getClipboardText()`가 "text bold text"를 반환하는 경우 `dreamweaver.getClipboardText(true)`는 "text bold text"를 반환합니다.

라이브러리 및 템플릿 함수

라이브러리 및 템플릿 함수는 문서와 템플릿 또는 라이브러리 항목 간에 링크를 만들고, 업데이트하고, 끊는 등 라이브러리 항목 및 템플릿과 관련된 작업을 처리합니다. `dreamweaver.libraryPalette` 객체의 메서드는 현재 문서가 아닌 [예셋] 패널 라이브러리 항목의 선택 영역을 제어하거나 선택 영역에 대한 작업을 수행합니다. 마찬가지로, `dreamweaver.templatePalette` 객체의 메서드는 [예셋] 패널 템플릿 객체의 선택 영역을 제어하거나 선택 영역에 대한 작업을 수행합니다.

dom.applyTemplate()

지원 버전

Dreamweaver 3

설명

현재 문서에 템플릿을 적용합니다. 인수를 제공하지 않으면 [템플릿 선택] 대화 상자가 나타납니다. 이 함수는 포커스를 가진 문서에서만 유효합니다.

인수

{`templateURL`}, `bMaintainLink`

- **templateURL** 인수는 현재 사이트에 있는 템플릿의 경로이며, `file://` URL로 표시됩니다.
- **bMaintainLink** 인수는 원본 템플릿과의 연결을 유지할지(`true`), 아니면 유지하지 않을지(`false`)를 나타내는 부울 값입니다.

반환값

없음

활성자

431페이지의 "[dom.canApplyTemplate\(\)](#)"을 참조하십시오.

dom.detachFromLibrary()

지원 버전

Dreamweaver 3

설명

선택 영역 주위에서 잠금 태그를 제거하여 관련 LBI 파일에서 선택된 라이브러리 항목 인스턴스를 떼어냅니다. 이 함수는 속성 관리자 내의 [원본에서 떼어내기]를 클릭하는 것과 동일합니다.

인수

없음

반환값

없음

dom.detachFromTemplate()

지원 버전

Dreamweaver 3

설명

현재 문서를 관련 템플릿에서 떼어냅니다.

인수

없음

반환값

없음

dom.getAttachedTemplate()

지원 버전

Dreamweaver 3

설명

문서와 연관된 템플릿의 경로를 가져옵니다.

인수

없음

반환값

file:// URL로 표시되는 템플릿의 경로를 포함하는 문자열을 반환합니다.

dom.getEditableRegionList()

지원 버전

Dreamweaver 3

설명

문서의 본문에서 모든 편집 가능 영역의 목록을 가져옵니다.

인수

없음

반환값

요소 노드의 배열을 반환합니다.

예제

289페이지의 “[dom.getSelectedEditableRegion\(\)](#)”

dom.getIsLibraryDocument()

지원 버전

Dreamweaver 3

설명

문서가 라이브러리 항목인지 여부를 확인합니다.

인수

없음

반환값

문서가 LBI 파일인지 여부를 나타내는 부울 값을 반환합니다.

dom.getIsTemplateDocument()

지원 버전

Dreamweaver 3

설명

문서가 템플릿인지 여부를 확인합니다.

인수

없음

반환값

문서가 DWT 파일인지 여부를 나타내는 부울 값을 반환합니다.

dom.getSelectedEditableRegion()

지원 버전

Dreamweaver 3

설명

선택 영역이나 삽입점이 편집 가능 영역 내에 있으면 이 함수는 문서의 본문에 있는 항목 중에서 편집 가능 영역의 위치를 가져옵니다.

인수

없음

반환값

dom.getEditableRegionList() 함수가 반환하는 배열의 인덱스를 반환합니다. 자세한 내용은 288페이지의 “dom.getEditableRegionList()”를 참조하십시오.

예제

다음 코드에서는 선택된 편집 가능 영역의 내용이 포함된 대화 상자를 보여 줍니다.

```
var theDOM = dw.getDocumentDOM();  
var edRegs = theDOM.getEditableRegionList();  
var selReg = theDOM.getSelectedEditableRegion();  
alert(edRegs[selReg].innerHTML);
```

dom.insertLibraryItem()

지원 버전

Dreamweaver 3

설명

라이브러리 항목의 인스턴스를 문서에 삽입합니다.

인수

libraryItemURL

- libraryItemURL 인수는 LBI 파일의 경로이며, file:// URL로 표시됩니다.

반환값

없음

dom.markSelectionAsEditable()

지원 버전

Dreamweaver 3

설명

[새 편집 가능 영역] 대화 상자를 표시합니다. 사용자가 [새 영역]을 클릭하면 선택 영역이 편집 가능으로 표시되고 텍스트는 변경되지 않습니다.

인수

없음

반환값

없음

활성자

437페이지의 “dom.canMarkSelectionAsEditable()”을 참조하십시오.

dom.newEditableRegion()

지원 버전

Dreamweaver 3

설명

[새 편집 가능 영역] 대화 상자를 표시합니다. 사용자가 [새 영역]을 클릭하면 영역의 이름이 중괄호({ })로 묶여 문서의 삽입점 위치에 삽입됩니다.

인수

없음

반환값

없음

활성자

437페이지의 “[dom.canMakeNewEditableRegion\(\)](#)”을 참조하십시오.

dom.removeEditableRegion()

지원 버전

Dreamweaver 3

설명

문서에서 편집 가능 영역을 제거합니다. 편집 가능 영역에 내용이 포함되어 있으면 해당 내용은 보존되고 편집 가능 영역 표시자만 제거됩니다.

인수

없음

반환값

없음

활성자

438페이지의 “[dom.canRemoveEditableRegion\(\)](#)”을 참조하십시오.

dom.updateCurrentPage()

지원 버전

Dreamweaver 3

설명

문서의 라이브러리 항목이나 템플릿 또는 둘 다 업데이트합니다. 이 함수는 활성 문서에만 사용할 수 있습니다.

인수**{typeOfUpdate}**

- 선택적인 **typeOfUpdate** 인수는 "library", "template" 또는 "both"여야 합니다. 인수를 생략하는 경우 기본값은 "both"입니다.

반환값

없음

dreamweaver.updatePages()

지원 버전

Dreamweaver 3

설명

[페이지 업데이트] 대화 상자를 열고 지정 한 옵션을 선택합니다.

인수**{typeOfUpdate}**

- 선택적인 **typeOfUpdate** 인수는 지정 하는 경우 "library", "template" 또는 "both"여야 합니다. 이 인수를 생략하면 기본값은 "both"입니다.

반환값

없음

코드 단편 패널 함수

Dreamweaver를 사용하면 웹 개발자는 [코드 단편] 패널에서 재사용 가능한 코드 블록을 편집 및 저장할 수 있으며 필요에 따라 가져올 수 있습니다.

[코드 단편] 패널은 각 코드 단편을 Configuration/Snippets 폴더의 CSN 파일에 저장합니다. Dreamweaver에 들어 있는 코드 단편은 다음 폴더에 저장되어 있습니다.

- Accessible
- Comments
- Content_tables
- Filelist.txt
- Footers
- Form_elements
- Headers
- Javascript
- Meta
- Navigation
- Text

코드 단편 파일은 XML 문서이므로 다음 예제와 같이 XML 지시문에 인코딩을 지정할 수 있습니다.

```
<?XML version="1.0" encoding="utf-8">
```

다음은 코드 단편 파일의 예제입니다.

```
<snippet name="Detect Flash" description="VBScript to check for Flash ActiveX control" preview="code"
factory="true" type="wrap" >
  <insertText location="beforeSelection">
    <![CDATA[ ----- code ----- ]]>
  </insertText>
  <insertText location="afterSelection">
    <![CDATA[ ----- code ----- ]]>
  </insertText>
</snippet>
```

CSN 파일의 **snippet** 태그에는 다음과 같은 속성이 있습니다.

속성	설명
name	코드 단편의 이름
description	코드 단편 설명
preview	미리 보기 유형: 코드 단편을 [미리 보기] 영역에 표시하는 경우 "code", 코드 단편을 [미리 보기] 영역에서 HTML로 렌더링하는 경우 "design"
type	사용자 선택 영역 앞뒤에 코드 단편을 넣는 경우 "wrap", 선택 영역 앞에 코드 단편을 삽입하는 경우 "block"

다음 메서드를 사용하면 코드 단편 패널 함수를 Extension에 추가할 수 있습니다.

dreamweaver.snippetPalette.getCurrentSnippetPath()

지원 버전

Dreamweaver MX 2004

설명

[코드 단편] 패널에서 현재 선택된 코드 단편의 경로를 반환합니다.

인수

없음

반환값

[코드 단편] 패널에서 선택된 코드 단편의 경로(Snippets 폴더에 상대적인 경로)를 반환합니다. 선택된 코드 단편이 없으면 빈 문자열을 반환합니다.

dreamweaver.snippetPalette.newFolder()

지원 버전

Dreamweaver MX

설명

기본 이름 **untitled**라는 이름의 새 폴더를 만들고 기본 이름 주위에 텍스트 상자를 그립니다.

인수
없음

반환값
없음

dreamweaver.snippetPalette.newSnippet()

지원 버전
Dreamweaver MX

설명
[코드 단편 추가] 대화 상자를 열고 포커스를 둡니다.

인수
없음

반환값
없음

dreamweaver.snippetPalette.editSnippet()

지원 버전
Dreamweaver MX

설명
[코드 단편 편집] 대화 상자를 열고 포커스를 두어 선택한 요소를 편집할 수 있게 합니다.

인수
없음

반환값
없음

활성자
458페이지의 “[dreamweaver.snippetpalette.canEditSnippet\(\)](#)”을 참조하십시오.

dreamweaver.snippetPalette.insert()

지원 버전
Dreamweaver MX

설명
[코드 단편] 패널에서 선택한 코드 단편을 현재 선택 영역에 적용합니다.

인수

없음

반환값

없음

활성자458페이지의 “[dreamweaver.snippetpalette.canInsert\(\)](#)”를 참조하십시오.

dreamweaver.snippetPalette.insertSnippet()

지원 버전

Dreamweaver MX

설명

지정된 코드 단편을 현재 선택 영역에 삽입합니다.

인수**path**

- Snippets 폴더에 상대적으로 코드 단편의 경로를 지정하는 문자열입니다.

반환값

부울 값을 반환합니다.

활성자458페이지의 “[dreamweaver.snippetpalette.canInsert\(\)](#)”를 참조하십시오.**예제**

다음과 같이 dw.snippetPalette.insertSnippet() 함수를 호출하면 인수로 지정된 위치의 코드 단편이 현재 문서의 삽입점에 삽입됩니다.

```
dw.snippetPalette.insertSnippet('Text\\Different_Link_Color.csn');
```

dreamweaver.snippetPalette.rename()

지원 버전

Dreamweaver MX

설명

선택한 폴더 이름 또는 파일 별칭 주위의 텍스트 상자를 활성화하여 선택한 요소를 편집할 수 있도록 합니다.

인수

없음

반환값

없음

dreamweaver.snippetPalette.remove()

지원 버전

Dreamweaver MX

설명

선택한 요소 또는 폴더를 [코드 단편] 패널에서 삭제하고 디스크에서 파일을 삭제합니다.

반환값

없음

Spry 위젯 편집 함수

Dreamweaver CS4에서는 Spry 및 기타 동적 위젯에 대한 향상된 편집 함수를 제공합니다.

element.getTranslatedAttribute()

지원 버전

Dreamweaver CS3

설명

이 함수는 W3C `getAttribute()` 함수와 동일하지만 변환된 속성에 대해 작동합니다. `element.getTranslatedAttribute()` 함수는 이름별로 속성 값을 가져옵니다.

인수

name

- **name** 인수는 가져올 속성의 이름인 DOMString입니다.

반환값

속성 이름을 DOMString으로 반환합니다. 속성에 지정된 값이나 기본값이 없는 경우에는 빈 문자열을 반환합니다.

element.removeTranslatedAttribute()

지원 버전

Dreamweaver CS3

설명

이 함수는 W3C `removeAttribute()` 함수와 동일하지만 변환된 속성에 대해 작동합니다. `element.removeTranslatedAttribute()` 함수는 이름별로 속성을 제거합니다. 속성에 기본값이 있는 경우에는 기본값과 해당 네임스페이스 URI, 로컬 이름 및 접두어(해당되는 경우)와 함께 속성이 표시됩니다.

인수

name

- **name** 인수는 제거할 속성의 이름인 DOMString입니다.

반환값
없음

element.setTranslatedAttribute()

지원 버전
Dreamweaver CS3

설명

이 함수는 W3C `setAttribute()` 함수와 동일하지만 변환된 속성에 대해 작동합니다. `element.setTranslatedAttribute()` 함수는 지정된 값을 가진 새로운 속성을 추가합니다. 요소에 지정된 이름을 가진 속성이 이미 존재하는 경우에는 속성 값이 **value** 인수에 지정된 값으로 변경됩니다.

value는 간단한 문자열이며, 설정된 경우 파싱되지 않습니다. 따라서 이 문자열에 포함된 모든 구문은 간단한 텍스트로 처리되며 작성 시에 구현을 통해 적절하게 이스케이프해야 합니다.

엔터티 참조로 인식되는 구문을 포함하는 속성 값을 할당하려면 `Attr` 노드를 모든 `Text` 및 `EntityReference` 노드와 함께 만들고 적절한 하위 트리를 구성한 후 `setAttributeNode`를 사용하여 속성 값으로 할당해야 합니다.

인수

name, value

- **name** 인수는 만들거나 변경할 속성의 이름인 `DOMString`입니다.
- **value** 인수는 속성에 설정할 값인 `DOMString`입니다.

반환값
없음

element.translatedClassName

지원 버전
Dreamweaver CS3

설명

이 함수는 `element.ClassName()` 함수와 동일하지만 변환된 `className` 속성에 대해 작동합니다.

element.translatedStyle

지원 버전
Dreamweaver CS3

설명

이 함수는 `element.style()` 함수와 동일하지만 변환된 `style` 속성에 대해 작동합니다.

예제

```
var div1 = dom.getElementById("div1");  
div1.translatedStyle.display = "none";
```

Spry 위젯 삽입 함수

Dreamweaver에서는 Spry 위젯을 손쉽게 삽입할 수 있도록 다음 함수를 제공합니다.

dom.addJavaScript()

지원 버전

Dreamweaver CS3

설명

이 함수는 Dreamweaver에서 헤드 또는 본문에 JavaScript 블록을 삽입하도록 합니다. 본문에 삽입하는 경우 JavaScript 블록이 </body> 태그 바로 앞에 삽입됩니다. 문서에 JavaScript 블록이 이미 있는 경우에는 새 <script> 태그가 삽입되지 않고 <script> 내용에 "code"가 추가됩니다.

인수

code, insideHead

- code 인수는 페이지에 삽입할 JavaScript 코드를 포함하는 문자열입니다.
- insideHead 인수는 JavaScript 블록을 헤드에 삽입할지 아니면 본문에 삽입할지를 나타내는 부울 값입니다. 기본값은 true로, 헤드에 코드를 삽입합니다. false인 경우 코드는 본문에서 </body> 태그 바로 앞에 삽입됩니다. 이 인수는 선택 사항입니다.

반환값

없음

예제

```
function objectTag()
{
    .
    .
    .
    var dom = dw.getDocumentDOM();
    var id = dwscripts.getUniqueId("accordion");
    var code = "new Accordion(' + id + ', 250, {duration:200, step:20})";
    dom.addJavaScript(code, false);

    return retVal;
}
```

dom.copyAssets()

지원 버전

Dreamweaver CS3, CS4에서 업데이트됨

설명

Extension 제작자는 이 API를 사용하여 사용자 사이트에 외부 종속 파일을 복사하고 페이지 헤드에 필수 파일 참조를 추가할 수 있습니다.

인수

assetArray

JavaScript 객체의 배열입니다. 각 JavaScript 객체에는 *srcURL*, *destURL*, *referenceType*, *useDefaultFolder* 및 *documentRelative* 필드가 있습니다.

- *srcURL* 인수는 예셋의 경로이며, file://URL로 표시됩니다.
- *destURL* 인수는 예셋을 복사할 위치를 지정하는 상대 경로입니다. *destURL*은 *useDefaultFolder* 값에 따라 상대적으로 지정됩니다. *useDefaultFolder*가 true이면 경로는 기본 Asset 폴더에 상대적이고, *useDefaultFolder*가 false이면 경로는 사이트 루트에 상대적입니다. 사이트가 정의되어 있지 않으면 경로는 문서에 상대적입니다. *useDefaultFolder* 설명을 참조하십시오.
- *referenceType* 인수는 Extension 제작자가 헤드에 파일 참조를 삽입하려는 경우에 필요합니다. *referenceType*에 유효한 값은 다음과 같습니다.
 - link - 외부 CSS 파일에 대한 LINK 태그 삽입
 - import - @import가 포함된 STYLE 태그 삽입
 - javascript - type=text/javascript가 포함된 SCRIPT 태그 삽입
 - vbscript - type=text/vbscript가 포함된 SCRIPT 태그 삽입
 - "" - 헤드에 참조 삽입 안 함
- *The useDefaultFolder* 인수는 *destURL*에 지정된 경로가 기본 Assets 폴더에 상대적인지 여부를 나타내는 부울 값입니다. 값이 이 속성이 설정되지 않았음을 의미하는 false이면 *destURL*이 사이트 루트에 상대적인 것으로 간주됩니다. 사이트가 정의되어 있지 않으면 *destURL*은 문서에 상대적인 것으로 간주됩니다. 이 인수의 기본값은 false입니다.
- *documentRelative* 인수는 부울 값입니다. 기본값은 false입니다. 이 매개 변수가 false이면 파일이 사이트에 저장될 때 예셋이 사이트 루트에 상대적인 *destURL*에 지정된 폴더에 복사됩니다. 값이 true이면 예셋이 문서에 상대적인 *destURL*에 지정된 경로에 복사됩니다.

반환값

file://URL로 표시된 문자열 배열을 반환합니다. 각 문자열은 스크립트나 링크 태그를 통해 문서의 헤드에 포함된 파일을 나타냅니다.

예제

```
function objectTag()
{
    .
    .
    .
    var dom = dw.getDocumentDOM();
    var assetList = new Array();
    var assetInfo = new AssetInfo("Objects/Ajax/Accordion.css",
                                  "Objects/Ajax/Accordion.css",
                                  "Accordion.css", "link");

    assetList.push(assetInfo);
    assetInfo = new AssetInfo("Objects/Ajax/Accordion.js", "Accordion.js",
                              "javascript");
    assetList.push(assetInfo);
    assetInfo = new AssetInfo("Objects/Ajax/Images", "Images", "");
    assetList.push(assetInfo);
    dom.copyAssets(assetList);
    return retVal;
}
```

dom.getDefaultAssetFolder()

지원 버전

Dreamweaver CS3

설명

문서의 기본 에셋 폴더를 가져옵니다.

인수

없음

반환값

기본 에셋 폴더 이름인 문자열을 반환합니다.

예제

```
function objectTag()  
{  
.  
.  
.  
var defaultAssetFolder = dom.getDefaultAssetFolder();  
.  
.  
.  
return retVal;  
}
```

브라우저 호환성 확인 함수

다음 함수를 사용하여 브라우저 렌더링 버그를 일으키는 HTML과 CSS 조합을 찾을 수 있으며(자세한 내용은 **Dreamweaver** 확장의 "브라우저 호환성 확인 문제 API" 장 참조), 다른 여러 유형의 **Extension**(예: 명령)에도 사용할 수 있습니다.

참고: 이러한 함수에서 반환하는 값은 현재 디자인 뷰에서 사용 중인 스타일을 나타냅니다. 함수가 문제 파일에서 브라우저 호환성 확인의 일부로 사용된 경우 **Dreamweaver**는 대상 브라우저에서 읽는 방식에 따라 자동으로 스타일을 필터링하지만(예를 들어, 대상 브라우저가 Internet Explorer 6 이전 버전인 경우 Star HTML을 사용하여 정의된 스타일을 고려함), 함수를 브라우저 호환성 확인 범위를 벗어나서 사용한 경우에는 필터링이 수행되지 않습니다.

elem.getComputedStyleProp()

지원 버전

Dreamweaver CS3

설명

CSS에 지정되어 있는 속성의 위치와 관계없이 지정된 요소를 렌더링하는 데 사용되는 지정된 CSS 속성의 값을 가져옵니다. 브라우저와 달리, 값에 "px"가 지정되지 않은 경우에도 길이가 픽셀로 보고됩니다.

인수

propName, pseudoElt

- **propName** 인수는 CSS 속성의 이름입니다(하이픈 대신 대소문자 혼용 방식 사용, 예: "font-size" 대신 "fontSize").
- **pseudoElt** 인수는 CSS 유사 요소 또는 null입니다(아무 것도 없는 경우).

반환값

속성의 계산 값을 포함하는 문자열을 반환합니다.

참고: 숫자 값도 문자열로 반환됩니다. 계산에 이러한 문자열 값을 사용하려면 해당 값을 `parseInt()` 또는 `parseFloat()`를 사용하여 숫자로 변환합니다.

예제

```
var dom = dw.getDocumentDOM();
var myDiv = dom.getElementsByTagName('myDiv');
var float = myDiv.getComputedStyleProp("float");
if (float == "left")
    alert("This div is floated left.");
```

window.getDeclaredStyle()

지원 버전

Dreamweaver CS3

설명

지정된 요소에 대해 선언된 CSS 스타일을 가져옵니다. 명시적으로 선언되지 않은 스타일은 정의되지 않으며, 계산된 픽셀 값 대신 스타일 시트에 선언된 실제 길이 값(예: 20%, .8em)을 지정한다는 점에서 `getComputedStyle()` 함수와는 다릅니다. `bGetInherited`가 `false`(기본값)인 경우, `getDeclaredStyle()`은 요소에 직접 적용되는 스타일만 가져오며 부모에서 상속한 스타일은 포함하지 않습니다.

인수

elt, pseudoElt, psuedoClassList, bGetInherited

- **elt** - 스타일 정보가 필요한 문서의 노드
- **pseudoElt** - CSS 유사 요소 또는 null(아무 것도 없는 경우)
- **psuedoClassList** - 유사 클래스의 목록(각각 공백으로 구분)으로 구성된 선택적 문자열
- **bGetInherited** - 조상에서 상속한 스타일을 포함할지 여부를 나타내는 선택적 부울 값(기본값: false)

반환값

이름으로 액세스할 수 있는 스타일 속성을 포함하는 읽기 전용 객체를 반환합니다.

예제

```
var dom = dw.getDocumentDOM();
var myDiv = dom.getElementById('myDiv');
var props = window.getComputedStyle(myDiv);
var marleft = "";
var units = "";
if (typeof(props.marginLeft) != "undefined"){
    marleft = props.marginLeft;
    units = marleft.replace(/\d+/, ""); // remove digits, leaving units
    alert(units); // should show %, px, pt, em, etc.
}
else
    alert("no margin-left property has been set for myDiv.");
```

dom.getMinDisplayWidth()

지원 버전

Dreamweaver CS3

설명

전체 내용을 표시하는 데 필요한 블록 수준 컨테이너의 최소 너비를 가져옵니다.

참고: `dom.minDisplayWidth()` 함수에서 반환하는 값보다 작은 값이 CSS를 사용하여 지정된 경우 컨테이너의 실제 너비는 더 작을 수 있습니다.

인수**container**

- **container**는 최소 너비가 필요한 포함하는 요소입니다.

반환값

지정된 컨테이너의 최소 표시 너비를 나타내는 정수를 픽셀 단위로 반환하거나, 요소가 컨테이너가 아니거나 요소의 최소 너비를 확인할 수 없는 경우에는 -1을 반환합니다.

예제

```
var dom = dw.getDocumentDOM();  
var myDiv = dom.getElementById('myDiv');  
var props = window.getComputedStyle(myDiv);  
var minW = dom.getMinDisplayWidth(myDiv);  
var setW = props.width;  
if (minW > setW)  
alert("Depending on the browser, your content will either be \n" +  
      "clipped, or the container will expand beyond its set width.");
```

dom.getBlockElements() elem.getBlockElements()

지원 버전

Dreamweaver CS3

설명

문서 또는 요소에서 고유의 또는 지정된 'block' 표시 값을 가진 자손을 검색합니다.

인수

없음

반환값

요소 노드의 배열을 반환합니다.

예제

```
[...]
var blocks = DOM.getBlockElements();
var dProps = null, children = null;
for (var i=0; i < blocks.length; i++){
    // get the declared styles so we can see whether width
    // or height have been specified explicitly
    dProps = window.getDeclaredStyle(blocks[i]);
    // if the block has children, border-left, and padding-bottom
    // but no width or height
    if (blocks[i].hasChildNodes() && |
        issueUtils.hasBorder(blocks[i],null,"left") &&
        (parseFloat(blocks[i].getComputedStyle("padding-bottom")) > 0) &&
        typeof(dProps.width) == "undefined" && typeof(dProps.height) == "undefined"){
        children = blocks[i].getBlockElements();
        var hasLayout = false;
        // loop through the block-level children to see if
        // any have width or height defined. width or height on any
        // of the children of the outer block will prevent the bug.
        for (var j=0; j < children.length; j++){
            dProps = window.getDeclaredStyle(children[j]);
            if (typeof(dProps.width) != "undefined" || typeof(dProps.height) !=
                "undefined"){
                hasLayout = true;
                break;
            }
        }
    }
    [...]
}
[...]
```

dom.getInlineElements() elem.getInlineElements()

지원 버전

Dreamweaver CS3

설명

문서 또는 요소에서 고유의 또는 지정된 'inline' 표시 값을 가진 자손을 검색합니다.

인수

없음

반환값

요소 노드의 배열을 반환합니다.

예제

```
[...]
var DOM = dw.getDocumentDOM();
var inEls = DOM.body.getInlineElements();
var next = null, prev = null, parent = null;
var props = null;

// look through all inline elements for replaced elements.
// if no replaced elements are found, don't bother going forward.
for (var i=0; i < inEls.length; i++){
    if (inEls[i].tagName == 'IMG' ||
        inEls[i].tagName == 'INPUT' ||
        inEls[i].tagName == 'TEXTAREA' ||
        inEls[i].tagName == 'SELECT' ||
        inEls[i].tagName == 'OBJECT'){
        // do something
    }
}
[...]
```

dom.getHeaderElements() elem.getHeaderElements()**지원 버전**

Dreamweaver CS3

설명

문서 또는 요소에서 머리글 태그(H1-H6)를 검색합니다.

인수

없음

반환값

요소 노드의 배열을 반환합니다.

예제

```
var DOM = dw.getDocumentDOM();
var headers = DOM.getHeaderElements();

for (var i=0; i < headers.length; i++){
    alert(headers[i].tagName);
}
```

dom.getListElements() elem.getListElements()**지원 버전**

Dreamweaver CS3

설명

문서 또는 요소에서 순차, 비순차 또는 정의 목록을 검색합니다.

인수

없음

반환값

요소 노드의 배열을 반환합니다.

예제

```
[...]
var DOM = dw.getDocumentDOM();
// get all the UL, OL, and DL elements in the document.
var lists = DOM.getListElements();
var props = null;
for (var i=0; i < lists.length; i++){
    props = window.getComputedStyle(lists[i]);
    if ((props.cssFloat == "left" || props.cssFloat == "right") && props.overflow == "auto"){
        // do something
    }
}
[...]
```

elem.isBlockElement()

지원 버전

Dreamweaver CS3

설명

요소에 고유의 또는 지정된 'block' 표시 값이 있는지 확인합니다.

인수

없음

반환값

객체가 블록 수준 요소인지 여부를 나타내는 부울 값을 반환합니다.

예제

```
[...]
var DOM = dw.getDocumentDOM();
var blocks = DOM.body.getBlockElements();
var next = null;
for (var i=0; i < blocks.length; i++){
    // next is the node right after blocks[i]
    next = blocks[i].nextSibling;
    // if next isn't null AND next is an element node AND next is a block element,
    // we've met the "second of two consecutive block elements" test.
    if (next && (next.nodeType == 1) && next.isBlockElement()){
        // do something
    }
}
[...]
```

elem.isInlineElement()

지원 버전

Dreamweaver CS3

설명

요소에 고유의 또는 지정된 'inline' 표시 값이 있는지 확인합니다.

인수

없음

반환값

객체가 인라인 요소인지 여부를 나타내는 부울 값을 반환합니다.

예제

```
[...]
var DOM = dw.getDocumentDOM();
var floats = issueUtils.getFloats(DOM.body);
var next = null;
for (var i=0; i < floats.length; i++){
    next = floats[i].nextSibling;
    // if nextSibling of float is a text node or an inline element
    if (next && (next.nodeType == Node.TEXT_NODE ||
        (next.nodeType == Node.ELEMENT_NODE && next.isInlineElement()))){
        // do something
    }
}
[...]
```

elem.isHeaderElement()

지원 버전

Dreamweaver CS3

설명

요소가 h1, h2, h3, h4, h5, h6 태그 중 하나인지 확인합니다.

인수

없음

반환값

객체가 머릿글 요소인지 여부를 나타내는 부울 값을 반환합니다.

예제

```
[...]
var DOM = dw.getDocumentDOM();
var floats = issueUtils.getFloats(DOM.body);
var prev = null;
// first float in the document isn't affected, so start
// at 1.
for (var i=1; i < floats.length; i++){
    prev = floats[i].previousSibling;
    // if the element before the float is a header
    if (prev && prev.isHeaderElement()){
        // do something
    }
}
[...]
```

elem.isListElement()

지원 버전

Dreamweaver CS3

설명

요소가 ul, ol, dl 태그 중 하나인지 확인합니다.

인수

없음

반환값

객체가 목록 요소인지 여부를 나타내는 부울 값을 반환합니다.

예제

```
[...]
var DOM = dw.getDocumentDOM();
var floats = issueUtils.getFloats(DOM.body);
var prev = null, children = null;
for (var i=0; i < floats.length; i++){
    children = floats[i].childNodes;
    for (var k=0; k < children.length; k++){
        if (children[k].isListElement()){
            // do something
        }
    }
}
[...]
```

16장: 동적 문서

Adobe® Dreamweaver®의 동적 문서 함수는 웹 서버 페이지와 관련된 작업을 수행합니다. 이러한 작업은 다음과 같습니다.

- [구성 요소] 패널에서 선택한 노드에 대한 속성 반환
- 사용자 문서에 있는 모든 데이터 소스 목록 가져오기
- 디자인 뷰에 동적 내용 표시
- 문서에 서버 비헤이비어 적용
- 현재 정의되어 있는 모든 서버 모델의 이름 가져오기

서버 구성 요소 함수

서버 구성 요소 함수를 사용하면 [구성 요소] 패널에 나타나는 서버 구성 요소 트리 컨트롤에서 현재 선택한 노드에 액세스할 수 있습니다. 이러한 함수를 사용하면 구성 요소 트리의 뷰를 새로 고칠 수도 있습니다.

`dreamweaver.serverComponents.getSelectedNode()`

지원 버전

Dreamweaver MX

설명

서버 구성 요소 트리 컨트롤에서 현재 선택한 ComponentRec 속성을 반환합니다.

인수

없음

반환값

ComponentRec 속성을 반환합니다.

`dreamweaver.serverComponents.refresh()`

지원 버전

Dreamweaver MX

설명

구성 요소 트리의 뷰를 새로 고칩니다.

인수

없음

반환값

없음

데이터 소스 함수

데이터 소스 파일은 Configuration/DataSources 폴더에 저장됩니다. 각 서버 모델에는 ASP.Net/C#, ASP.Net/VisualBasic, ASP/JavaScript, ASP/VBScript, ColdFusion, JSP 및 PHP/MySQL과 같은 고유한 폴더가 있습니다. 각 서버 모델 하위 폴더에는 해당 서버 모델의 데이터 소스와 연관된 HTML 및 EDML 파일이 들어 있습니다.

Dreamweaver에서 데이터 소스를 사용하는 방법에 대한 자세한 내용은 **Dreamweaver** 확장의 "데이터 소스"를 참조하십시오.

dreamweaver.dbi.getDataSources

지원 버전

Dreamweaver UltraDev 4

설명

Configuration/DataSources 폴더의 각 파일에 대해 findDynamicSources() 함수를 호출합니다. 이 함수를 사용하여 사용자 문서의 모든 데이터 소스 목록을 생성할 수 있습니다. 이 함수는 Configuration/DataSources 폴더의 모든 파일에 반복 적용되어 각 파일에서 findDynamicSources() 함수를 호출한 다음 반환되는 배열을 모두 연결하고 연결된 데이터 소스의 배열을 반환합니다.

인수

없음

반환값

사용자 문서의 모든 데이터 소스가 연결된 목록을 포함하는 문자열을 반환합니다. 배열의 각 요소는 객체이며, 각 객체에는 다음과 같은 속성이 있습니다.

- **title** 속성은 각 부모 노드 아이콘의 오른쪽에 나타나는 레이블 문자열입니다. **title** 속성은 항상 정의됩니다.
- **imageFile** 속성은 [동적 데이터] 대화 상자, [동적 텍스트] 대화 상자 또는 [바인딩] 패널에서 부모 노드를 나타내는 아이콘 (GIF 이미지)을 포함하는 파일의 경로입니다. **imageFile** 속성은 항상 정의됩니다.
- **allowDelete** 속성은 선택 사항입니다. 이 속성을 **false**로 설정하면 사용자가 [바인딩] 패널에서 이 노드를 클릭할 때 마이너스 (-) 버튼이 비활성화되고, **true**로 설정하면 마이너스(-) 버튼이 활성화됩니다. 이 속성이 정의되지 않은 경우에는 속성이 **true**로 설정된 것처럼 사용자가 항목을 클릭할 때 마이너스(-) 버튼이 활성화됩니다.
- **dataSource** 속성은 findDynamicSources() 함수가 정의되어 있는 파일의 간단한 이름입니다. 예를 들어, Configuration/DataSources/ASP_Js 폴더의 Session.htm 파일에서 findDynamicSources() 함수는 dataSource 속성을 session.htm으로 설정합니다. 이 속성은 항상 정의됩니다.
- **name** 속성은 데이터 소스인 dataSource가 존재할 경우 이와 연관된 서버 비헤이비어의 이름입니다. **name** 속성은 항상 정의되지만 세션 변수와 같이 데이터 소스와 연관된 서버 비헤이비어가 없는 경우 빈 문자열("")이 될 수도 있습니다.

dw.dbi.setExpanded()

지원 버전

Dreamweaver CS3

설명

확장 또는 축소할 노드를 설정합니다.

인수**data-source-node-name, expanded**

- **data-source-node-name**은 확장 또는 축소할 데이터 소스의 이름을 나타내는 문자열입니다.
- **expanded**는 데이터 세트 노드를 확장할지, 아니면 축소할지를 나타내는 부울 값입니다.

반환값

없음

예제

```
dw.dbi.setExpanded(dsName, true);           //expand the data source node
```

Extension Data Manager 함수

이 단원의 API는 EDM(Extension Data Manager)을 구성합니다. 이러한 함수를 호출하여 그룹과 참여자 파일에 포함된 데이터를 프로그래밍 방식으로 액세스하고 조작할 수 있습니다. EDM은 다음과 같은 방법으로 작업을 수행합니다.

- EDM은 그룹과 참여자 파일에 대해 모든 EDML 파일 입/출력을 수행합니다.
- EDM은 현재 서버 모델에 대한 모든 데이터 요청을 수행하여 서버 모델 필터로 작용합니다.

dreamweaver.getExtDataValue()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 지정된 노드에 대한 EDML 파일에서 필드 값을 가져옵니다.

인수**qualifier(s)**

- **qualifier(s)** 인수는 그룹 또는 참여자 이름, 하위 블록(있는 경우) 및 필드 이름을 포함하는 쉼표로 구분된 노드 한정자의 가변 길이 목록(필요한 정보 수준에 따라 다름)입니다.

반환값

필드 값을 반환합니다. 값이 지정되어 있지 않으면 기본값이 사용됩니다.

예제

다음 예제에서는 recordset_main 참여자의 insertText 태그에 대한 location 속성 값을 가져옵니다.

```
dw.getExtDataValue("recordset_main", "insertText", "location");
```

dreamweaver.getExtDataArray()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 지정된 노드에 대한 EDML 파일에서 값 배열을 가져옵니다.

인수**qualifier(s)**

- **qualifier(s)** 인수는 그룹 또는 참여자 이름, 하위 블록(있는 경우) 및 필드 이름을 포함하는 쉼표로 구분된 노드 한정자의 가변 길이 목록입니다.

반환값

자식 노드 이름의 배열을 반환합니다.

dreamweaver.getExtParticipants()

지원 버전

Dreamweaver UltraDev 4

설명

이 함수는 EDML 그룹 파일 또는 참여자 파일에서 참여자 목록을 가져옵니다.

인수**value, qualifier(s)**

- **value** 인수는 속성 값이며, 공백이면 무시됩니다. 예를 들어, `dreamweaver.getExtParticipants("", "participant");`입니다.
- **qualifier(s)** 인수는 필수 속성의 쉼표로 구분된 노드 한정자의 가변 길이 목록입니다.

반환값

지정된 속성을 가진 참여자 이름의 배열을 반환하며, 속성은 지정된 값과 일치해야 합니다.

dreamweaver.getExtGroups()

지원 버전

Dreamweaver UltraDev 4

설명

EDML 그룹 파일에서 서버 비헤이비어의 이름과 같은 그룹의 이름을 가져옵니다.

인수**value, qualifier(s)**

- **value** 인수는 속성 값이거나 무시할 수 있는 공백입니다.
- **qualifier(s)** 인수는 필수 속성의 쉼표로 구분된 노드 한정자의 가변 길이 목록입니다.

반환값

지정된 속성을 가진 그룹 이름의 배열을 반환하며, 속성은 지정된 값과 일치해야 합니다.

dreamweaver.refreshExtData()

지원 버전

Dreamweaver UltraDev 4

설명

모든 Extension 데이터 파일을 다시 로드합니다.



이 함수를 사용하면 Dreamweaver를 재시작하지 않고도 서버 비헤이비어 EDML 파일을 편집하여 다시 로드할 수 있는 유용한 명령을 만들 수 있습니다.

인수

없음

반환값

다시 로드된 데이터를 반환합니다.

라이브 데이터 함수

다음 라이브 데이터 함수를 사용하여 메뉴 기능을 모방할 수 있습니다.

- showLiveDataDialog() 함수는 [보기] > [라이브 데이터 설정] 메뉴 항목을 모방하는 데 사용됩니다.
- setLiveDataMode() 함수는 [보기] > [라이브 데이터] 및 [보기] > [라이브 데이터 새로 고침] 메뉴 항목을 모방하는 데 사용됩니다.
- getLiveDataMode() 함수는 라이브 데이터 모드가 활성화되어 있는지 여부를 확인합니다.

변환기 API liveDataTranslateMarkup() 함수를 구현할 때 나머지 라이브 데이터 함수를 사용할 수 있습니다.

dreamweaver.getLiveDataInitTags()

지원 버전

Dreamweaver UltraDev 1

설명

현재 활성 문서의 초기화 태그를 반환합니다. 초기화 태그는 사용자가 [라이브 데이터 설정] 대화 상자에 입력하는 HTML 태그입니다. 이 함수는 변환기가 liveDataTranslate() 함수에 태그를 전달할 수 있도록 일반적으로 변환기의 liveDataTranslateMarkup() 함수에서 호출됩니다.

인수

없음

반환값

초기화 태그를 포함하는 문자열을 반환합니다.

dreamweaver.getLiveDataMode()

지원 버전

Dreamweaver UltraDev 1

설명

[라이브 데이터] 윈도우가 현재 표시되는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [라이브 데이터] 윈도우가 표시되면 true를 반환하고, 그렇지 않으면 false를 반환합니다.

dreamweaver.getLiveDataParameters ()

지원 버전

Dreamweaver MX

설명

라이브 데이터 설정으로 지정된 URL 매개 변수를 얻습니다.

라이브 데이터 모드를 사용하면 응용 프로그램 서버에서 웹 페이지를 변환하여 반환한 것처럼 디자인 단계에서 웹 페이지를 볼 수 있습니다. [디자인] 뷰에 표시할 동적 내용을 생성하면 라이브 데이터가 있는 페이지 레이아웃을 볼 수 있으며 필요한 경우 이를 조정할 수 있습니다.

라이브 데이터를 보려면 문서에서 참조하는 모든 URL 매개 변수에 대해 라이브 데이터 설정을 입력해야 합니다. 이렇게 하면 웹 서버가 시뮬레이션에서 정의되지 않은 매개 변수에 대한 오류를 반환하는 것을 막을 수 있습니다.

이름-값 쌍에 URL 매개 변수를 입력합니다. 예를 들어, 문서의 서버 스크립트에서 URL 변수 ID 및 Name을 참조하는 경우 이러한 URL 매개 변수를 설정해야 라이브 데이터를 볼 수 있습니다.

Dreamweaver에서는 다음과 같은 방법으로 라이브 데이터 설정을 입력할 수 있습니다.

- [보기] 메뉴에서 활성화할 수 있는 [라이브 데이터 설정] 대화 상자를 통해.
- 툴바에서 [라이브 데이터 뷰] 버튼을 클릭할 때 문서의 맨 위에 나타나는 [URL] 텍스트 필드에서.

ID 및 Name 매개 변수에 대해 다음과 같은 쌍을 입력할 수 있습니다.

ID	22
Name	Samuel

이러한 매개 변수는 URL에 다음 예제와 같이 나타나게 됩니다.

`http://someURL?ID=22&Name=Samuel`

이 함수를 사용하면 JavaScript를 통해 이러한 라이브 데이터 설정을 얻을 수 있습니다.

인수

없음

반환값

현재 문서의 URL 매개 변수를 포함하는 배열을 반환합니다. 이 배열에는 매개 변수 문자열이 짝수 개 포함됩니다. 각각의 두 요소가 URL 매개 변수 이름-값 쌍을 구성합니다. 짝수 요소는 매개 변수 이름이며 홀수 요소는 해당 값입니다. 예를 들어, 위 예제에서 `getLiveDataParameters()`는 ID 및 Name 매개 변수에 대해 `['ID','22','Name','Samuel']` 배열을 반환합니다.

예제

다음 예제에서는 라이브 데이터 설정으로 지정된 매개 변수를 반환하여 `paramsArray`에 저장합니다.

```
var paramsArray = dreamweaver.getLiveDataParameters();
```

dreamweaver.liveDataTranslate()

지원 버전

Dreamweaver UltraDev 1

설명

응용 프로그램 서버에 전체 HTML 문서를 전송하고 문서의 스크립트를 실행하도록 서버에 요청한 다음 결과로 만들어진 HTML 문서를 반환합니다. 이 함수는 변환기의 `liveDataTranslateMarkup()` 함수에서만 호출할 수 있습니다. 다른 곳에서 호출하려고 하면 오류가 발생합니다. `dreamweaver.liveDataTranslate()` 함수는 다음 작업을 수행합니다.

- [라이브 데이터] 윈도우의 오른쪽 가장자리 근처에 나타나는 애니메이션 이미지를 재생합니다.
- 사용자 입력을 기다립니다. [중단] 아이콘을 클릭하면 이 함수는 즉시 반환됩니다.
- 호출자에서 단일 문자열 인수를 받습니다. 일반적으로 이 문자열은 사용자 문서의 전체 소스 코드입니다. 이 문자열은 다음 작업에서 사용되는 문자열과 같습니다.
- 사용자 문서의 HTML 문자열을 라이브 데이터 서버에 임시 파일로 저장합니다.
- [라이브 데이터 설정] 대화 상자에 지정된 매개 변수를 사용하여 HTTP 요청을 라이브 데이터 서버에 전송합니다.
- 라이브 데이터 서버로부터 HTML 응답을 받습니다.
- 라이브 데이터 서버에서 임시 파일을 제거합니다.
- 애니메이션 이미지 재생을 중단합니다.
- HTML 응답을 호출자에게 반환합니다.

인수

string

- 일반적으로 사용자의 현재 문서에 대한 전체 소스 코드인 단일 문자열입니다.

반환값

`httpReply` 객체를 반환합니다. 이 객체는 `MMHttp.getText()` 함수가 반환하는 값과 동일합니다. 사용자가 [중단] 아이콘을 클릭하는 경우 반환값의 `httpReply.statusCode` 값은 200(정상 상태)과 같으며 `httpReply.data` 값은 빈 문자열과 같습니다. `httpReply` 객체에 대한 자세한 내용은 16페이지의 “[HTTP API](#)”를 참조하십시오.

dreamweaver.setLiveDataError()

지원 버전

Dreamweaver UltraDev 1

설명

`liveDataTranslateMarkup()` 함수가 변환기에서 실행되는 동안 오류가 발생하는 경우 표시되는 오류 메시지를 지정합니다. Dreamweaver에서 `liveDataTranslate()`에 전달한 문서에 오류가 포함된 경우 서버는 HTML을 사용하여 서식을 지정한 오류 메시지를 반환합니다. 서버에서 오류 메시지를 반환했음을 변환기(`liveDataTranslate()`를 호출한 코드)가 확인하면 변환기는 `setLiveDataError()`를 호출하여 Dreamweaver에서 해당 오류 메시지를 표시합니다. 이 메시지는 `liveDataTranslateMarkup()` 함수의 실행이 끝난 후 표시됩니다. 오류 대화 상자에는 설명이 표시됩니다. `setLiveDataError()` 함수는 `liveDataTranslateMarkup()` 함수에서만 호출해야 합니다.

인수

source

- source** 인수는 오류 대화 상자에서 파싱하고 렌더링하는 소스 코드를 포함하는 문자열입니다.

반환값

없음

dreamweaver.setLiveDataMode()

지원 버전

Dreamweaver UltraDev 1

설명

[라이브 데이터] 윈도우의 가시성을 전환합니다.

인수

bIsVisible

- bIsVisible** 인수는 [라이브 데이터] 윈도우가 표시되는지 여부를 나타내는 부울 값입니다. 현재 [라이브 데이터] 윈도우가 표시된 경우 이 함수에 `true`를 전달하면 [새로 고침] 버튼을 클릭한 것과 동일한 효과가 있습니다.

반환값

없음

dreamweaver.setLiveDataParameters()

지원 버전

Dreamweaver MX

설명

라이브 데이터 모드에서 사용하기 위해 문서에서 참조하는 URL 매개 변수를 설정합니다.

라이브 데이터 모드를 사용하면 응용 프로그램 서버에서 웹 페이지를 변환하여 반환한 것처럼 디자인 단계에서 웹 페이지를 볼 수 있습니다. [디자인] 뷰에 표시할 동적 내용을 생성하면 라이브 데이터가 있는 페이지 레이아웃을 볼 수 있으며 필요한 경우 이를 조정할 수 있습니다.

라이브 데이터를 보려면 문서에서 참조하는 모든 URL 매개 변수에 대해 라이브 데이터 설정을 입력해야 합니다. 이렇게 하면 웹 서버가 시뮬레이션에서 정의되지 않은 매개 변수에 대한 오류를 반환하는 것을 막을 수 있습니다.

이름-값 쌍에 URL 매개 변수를 입력합니다. 예를 들어, 문서의 서버 스크립트에서 URL 변수 ID 및 Name을 참조하는 경우 이러한 URL 매개 변수를 설정해야 라이브 데이터를 볼 수 있습니다.

이 함수를 사용하면 JavaScript를 통해 라이브 데이터를 설정할 수 있습니다.

인수

liveDataString

- **liveDataString** 인수는 설정할 URL 매개 변수를 포함하는 이름-값 쌍 문자열입니다.

반환값

없음

예제

```
dreamweaver.setLiveDataParameters("ID=22&Name=Samuel")
```

dreamweaver.showLiveDataDialog()

지원 버전

Dreamweaver UltraDev 1

설명

[라이브 데이터 설정] 대화 상자를 표시합니다.

인수

없음

반환값

없음

라이브 뷰 함수

라이브 뷰 함수는 다음과 같은 용도로 사용됩니다.

- 디자인 뷰 모드 가져오기 및 설정
- 서버를 사용하여 라이브 뷰 모드 가져오기 및 설정
- 라이브 뷰 기본값 가져오기
- 라이브 뷰 종속 항목 가져오기 및 설정
- 라이브 뷰 매개 변수 보기

dom.getDesignViewMode()

지원 버전

Dreamweaver CS4

설명

이 함수는 디자인 뷰의 모드 또는 뷰를 가져옵니다. 디자인 뷰는 편집 가능한 클래식 디자인 뷰 또는 라이브 뷰에 있을 수 있습니다.

인수

없음

반환값

문자열 값을 반환합니다. 디자인 뷰의 모드가 라이브 뷰이면 **live**를 반환하고, 디자인 뷰의 모드가 편집 가능한 클래식 디자인 뷰이면 **editable**을 반환합니다.

dom.setDesignViewMode()

지원 버전

Dreamweaver CS4

설명

이 함수는 디자인 뷰의 모드를 활성화합니다. 예를 들어, 이 함수는 라이브 뷰를 활성화합니다.

인수

mode

- **mode** 인수는 **live** 또는 **editable** 값을 갖는 문자열입니다.

반환값

없음

dom.getLiveViewUsingServer()

지원 버전

Dreamweaver CS4

설명

이 함수를 사용하면 서버를 사용하여 현재 페이지를 미리 보는지 여부를 알 수 있습니다.

인수

없음

반환값

부울 값을 반환합니다. 서버를 사용하여 페이지를 미리 볼 수 있으면 **true**를 그렇지 않으면 **false**를 반환합니다.

dom.setLiveViewUsingServer()

지원 버전

Dreamweaver CS4

설명

이 함수를 사용하면 서버를 사용하여 페이지를 미리 볼 수 있는지 여부를 지정할 수 있습니다.

인수

bool

- 서버를 사용하여 페이지를 미리 볼 수 있는지 여부를 나타내는 bool 인수입니다. 이 함수에 **true**를 전달하면 서버를 사용하여 페이지를 미리 볼 수 있습니다.

반환값

없음

dom.getLiveViewDefaultsToUsingServer()

지원 버전

Dreamweaver CS4

설명

이 함수는 기본 작업이 서버를 사용하여 미리 보는 것인지 여부를 확인하는 데 사용됩니다.

인수

없음

반환값

부울 값을 반환합니다. 기본 작업이 서버를 사용하여 페이지를 미리 보는 것이면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.getLiveViewDependentsUsingServer()

지원 버전

Dreamweaver CS4

설명

이 함수는 종속 CSS 및 JavaScript 파일이 서버에서 요청되는지 여부를 확인하는 데 사용됩니다.

인수

없음

반환값

부울 값을 반환합니다. 종속 CSS 및 JavaScript 파일이 서버에서 요청되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.setLiveViewDependentsUsingServer()

지원 버전

Dreamweaver CS4

설명

이 함수는 종속 CSS 및 JavaScript 파일이 서버에서 요청되어야 하는지 여부를 지정하는 데 사용됩니다.

인수

bool

- 종속 CSS 및 JavaScript 파일이 서버에서 요청되는지 여부를 나타내는 부울 값입니다. 이 함수에 true를 전달하면 파일이 서버에서 요청됩니다.

반환값

없음

dom.showLiveViewParametersDialog()

지원 버전

Dreamweaver CS4

설명

이 함수는 라이브 미리 보기 매개 변수 대화 상자를 보는 데 사용됩니다.

인수

없음

반환값

없음

dom.browser.getSelection()

지원 버전

Dreamweaver CS4

설명

이 함수는 라이브 뷰에서 브라우저의 현재 선택 영역을 가져오는 데 사용됩니다.

인수

없음

반환값

소스 코드에 있는 선택 영역의 시작 및 끝 위치에 대한 두 오프셋이 포함된 배열을 반환합니다.

dom.browser.getStatusText()

지원 버전

Dreamweaver CS4

설명

이 함수는 브라우저의 현재 상태 텍스트를 가져오는 데 사용됩니다. 이 텍스트는 대개 빈 문자열이거나 "loading file ..." 문자열입니다.

인수

없음

반환값

브라우저 크롬의 상태 영역에 표시되는 텍스트를 반환합니다.

dom.browser.getWindow()

지원 버전

Dreamweaver CS4

설명

이 함수는 브라우저 컨트롤의 window 객체를 가져오는 데 사용됩니다. window 객체에서 브라우저 문서 객체 모델에 액세스할 수 있습니다.

인수

없음

반환값

객체를 반환합니다.

browserEle.loadHTML()

지원 버전

Dreamweaver CS4

설명

이 함수는 HTML 문자열을 브라우저 컨트롤에 직접 로드합니다. 이 함수는 HTML 문자열을 제공할 준비가 된 경우 유용합니다. HTML 문자열은 문서와 연결되지 않아야 합니다.

인수

없음

반환값

없음

dom.browser.interactivityPaused

지원 버전

Dreamweaver CS4

설명

이 속성을 사용하면 대화형 작업을 사용할 수 있는지 여부를 알 수 있습니다.

인수

없음

반환값

부울 값을 반환합니다. 대화형 작업을 사용할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.browser.javascriptEnabled

지원 버전

Dreamweaver CS4

설명

이 속성을 사용하면 JavaScript를 사용할 수 있는지 여부를 알 수 있습니다. 이 속성은 dom.interactivityPaused()와 같이 작동하지만 JavaScript에 대해서만 작동합니다.

인수

없음

반환값

부울 값을 반환합니다. JavaScript를 사용할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

<mm:browsercontrol>

확장 가능한 대화 상자에서 이 태그를 사용하여 대화 상자에서 브라우저를 표시할 수 있습니다. 이 태그에는 특수 속성이 없습니다. CSS 스타일을 사용하여 브라우저 윈도우의 크기를 제어할 수 있습니다. 태그에서 반환되는 DOM 객체는 dom.browser와 같은 유형의 객체이지만 동일한 인스턴스가 아닙니다. dom.browser 함수는 명령에서 작동하지 않으므로 DOM에서 브라우저 객체를 가져와야 합니다. 태그의 예는 다음과 같습니다.

```
<mm:browsercontrol id="myBrowser" style="width: 500px; height:300px;" />
```

문서의 헤드에 다음과 같은 스크립트가 있어야 합니다.

```
var browserEle = document.getElementById("myBrowser");
alert(browserEle.getWindow().document.documentElement.outerHTML);
```

브라우저 컨트롤은 다음 두 가지 특수 이벤트도 브로드캐스트합니다.

BrowserControlLoad 이 이벤트는 사용자가 로드된 브라우저 DOM에 사용자 고유의 요소를 첨부할 수 있도록 브라우저에서 로드 이벤트를 호출한 직후에 호출됩니다.

BrowserControlBeforeNavigation 이 이벤트는 브라우저에서 새 페이지로 이동하려고 할 때 호출됩니다. 이 이벤트가 취소되면 내비게이션 요청도 취소되고 브라우저 컨트롤이 현재 페이지에 유지됩니다. 이벤트 컨텍스트에도 요청된 URL이 포함되어 있습니다.

다음 예제에서는 이러한 이벤트의 기능을 보여 줍니다.

```
var browserEle = document.getElementById("myBrowser");
browserEle.addEventListener("BrowserControlBeforeNavigation",
    function(e){ if (e.requestedLocation = "foo.com")
        e.preventDefault(); //don't allow navigation to this site!}, true);
```

서버 비헤이비어 함수

서버 비헤이비어 함수를 사용하면 [윈도우] > [서버 비헤이비어]를 선택하여 표시할 수 있는 [서버 비헤이비어] 패널을 조작할 수 있습니다. 이러한 함수를 사용하면 페이지에서 모든 서버 비헤이비어를 찾고 프로그래밍 방식으로 문서에 새 비헤이비어를 적용하거나 기존 비헤이비어를 수정할 수 있습니다.

참고: `dw.serverBehaviorInspector`는 `dw.sbi`로 줄여 쓸 수 있습니다.

dreamweaver.getParticipants()

지원 버전

Dreamweaver UltraDev 4

설명

JavaScript 함수 `dreamweaver.getParticipants()`는 사용자의 문서에서 참여자 목록을 가져옵니다. `Dreamweaver`는 비헤이비어의 참여자를 모두 찾으면 해당 목록을 저장합니다. 일반적으로 이 함수를 `findServerBehaviors()` 함수(자세한 내용은 `Dreamweaver` 확장의 "서버 비헤이비어" 참조)와 함께 사용하여 사용자의 문서에서 비헤이비어 인스턴스를 찾습니다.

인수

`edmlFilename`

- **edmlFilename** 인수는 사용자의 문서에서 찾을 참여자의 이름이 포함된 그룹 또는 참여자 파일의 이름입니다. 이 문자열은 확장명 `.edml`이 없는 파일 이름입니다.

반환값

이 함수는 사용자의 문서에 나타나는 지정된 참여자의 모든 인스턴스 또는 그룹 파일의 경우 그룹에 있는 임의의 참여자의 인스턴스가 포함된 배열을 반환합니다. 배열은 JavaScript 객체로 구성되며, 사용자의 문서에서 찾을 수 있는 각 참여자의 각 인스턴스가 배열의 한 요소가 됩니다. 배열은 참여자가 문서에 나타나는 순서대로 정렬됩니다. 각 JavaScript 객체는 다음 속성을 갖습니다.

- **participantNode** 속성은 사용자 문서의 참여자 노드에 대한 포인터입니다.
- **participantName** 속성은 참여자의 EDML 파일 이름으로, 확장명 `.edml`은 표시되지 않습니다.
- **parameters** 속성은 모든 매개 변수/값 쌍을 저장하는 JavaScript 객체입니다.
- **matchRangeMin** 속성은 문서의 참여자 노드에서 참여자 내용의 시작 부분까지의 문자 오프셋을 정의합니다.
- **matchRangeMax** 속성은 참여자 노드의 시작 부분에서 참여자 내용의 마지막 문자까지의 오프셋을 정의하는 참여자의 정수입니다.

dreamweaver.serverBehaviorInspector.getServer Behaviors()

지원 버전

Dreamweaver UltraDev 1

설명

페이지에서 모든 비헤이비어 목록을 가져옵니다. `Dreamweaver`에서 서버 비헤이비어의 내부 목록을 업데이트할 필요가 있다고 판단하면 현재 설치된 각 비헤이비어에 대해 `findServerBehaviors()` 함수를 호출합니다. 각 함수는 배열을 반환합니다. 모든 배열이 하나의 배열로 병합된 후 각 비헤이비어의 `selectedNode` 객체가 문서에 나타나는 순서를 기준으로 정렬됩니다. 병합된 배열은 내부적으로 저장됩니다. `getServerBehaviors()` 함수는 병합된 배열에 대한 포인터를 반환합니다.

인수
없음

반환값

JavaScript 객체의 배열을 반환합니다. `findServerBehaviors()`를 호출하면 객체가 배열로 반환됩니다. 객체는 [서버 비헤이비어] 패널에 표시되는 순서대로 정렬됩니다.

dreamweaver.popupServerBehavior()

지원 버전

Dreamweaver UltraDev 1

설명

새로운 서버 비헤이비어를 문서에 적용하거나 기존 비헤이비어를 수정합니다. 사용자가 비헤이비어의 매개 변수를 지정해야 하는 경우에는 대화 상자가 나타납니다.

인수

{behaviorName} 또는 {behaviorObject}

- 선택적인 **behaviorName** 인수는 비헤이비어의 이름, 파일의 **title** 태그 또는 파일 이름을 나타내는 문자열입니다.
- 선택적인 **behaviorObject** 인수는 비헤이비어 객체입니다.

인수를 생략하는 경우 현재 선택한 서버 비헤이비어가 실행됩니다. 인수가 서버 비헤이비어의 이름인 경우 페이지에 비헤이비어가 추가됩니다. 인수가 `getServerBehaviors()` 함수에서 반환되는 배열의 객체 중 하나인 경우에는 대화 상자가 표시되므로 사용자가 비헤이비어의 매개 변수를 수정할 수 있습니다.

반환값

없음

서버 모델 함수

Dreamweaver에서는 문서마다 연관된 문서 형식이 있습니다. 동적 문서 형식의 경우 ASP-JS, ColdFusion 또는 PHP-MySQL과 같은 서버 모델도 연관됩니다.

서버 모델은 서버 기술과 관련된 기능을 그룹화하는 데 사용됩니다. 서로 다른 서버 비헤이비어, 데이터 소스 등이 문서와 연관된 서버 모델을 기준으로 나타납니다.

서버 모델 함수를 사용하면 현재 정의되어 있는 서버 모델 집합, 현재 서버 모델의 이름, 언어 및 버전, 현재 서버 모델이 해당 이름의 문자 세트(예: UTF-8)를 지원하는지 여부를 확인할 수 있습니다.

참고: Dreamweaver에서는 서버 모델 HTML 파일의 모든 정보를 읽고 서버 모델을 맨 처음 로드할 때 이 정보를 저장합니다. 따라서 `Extension`이 `dom.serverModel.getServerName()`, `dom.serverModel.getServerLanguage()` 및 `dom.serverModel.getServerVersion()`과 같은 함수를 호출할 때 이러한 함수는 저장된 값을 반환합니다.

dom.serverModel.getAppURLPrefix()

지원 버전

Dreamweaver MX

설명

테스트 서버에 있는 사이트의 루트 폴더 URL을 반환합니다. 이 URL은 [사이트 정의] 대화 상자의 [고급] 탭 아래에 있는 [테스트 서버]에 지정된 것과 동일합니다.

Dreamweaver는 사용자의 테스트 서버와 통신할 때 브라우저와 마찬가지로 HTTP를 사용합니다. 이때 이 URL을 사용하여 사이트의 루트 폴더에 액세스합니다.

인수

없음

반환값

라이브 데이터 및 디버그에 사용되는 응용 프로그램 서버의 URL을 포함하는 문자열을 반환합니다.

예제

사용자가 사이트를 만든 다음 테스트 서버가 로컬 컴퓨터에 있다고 지정하고 루트 폴더의 이름을 "employeeapp"로 지정한 경우 dom.serverModel.getAppURLPrefix() 함수를 호출하면 http://localhost/mycoolapp/ 문자열이 반환됩니다.

dom.serverModel.getDelimiters()

지원 버전

Dreamweaver MX

설명

JavaScript 코드가 각 서버 모델의 스크립트 구분 기호를 가져올 수 있도록 함으로써 서버 모델 코드 관리를 사용자 스크립트 코드 관리와 구분합니다.

인수

없음

반환값

다음 세 속성을 각각 포함하는 객체의 배열을 반환합니다.

- **startPattern** 속성은 열기 스크립트 구분 기호를 찾는 정규식입니다.
- **endPattern** 속성은 닫기 스크립트 구분 기호를 찾는 정규식입니다.
- **participateInMerge** 패턴은 목록의 구분 기호로 둘러싸인 내용이 블록 병합에 참여하는지 여부를 지정하는 부울 값입니다. 이 경우, 참여할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.serverModel.getDisplayName()

지원 버전

Dreamweaver MX

설명

UI(사용자 인터페이스)에 나타나는 서버 모델의 이름을 가져옵니다.

인수

없음

반환값

서버 모델의 이름을 나타내는 문자열을 반환합니다.

dom.serverModel.getFolderName()

지원 버전

Dreamweaver MX

설명

ServerModels 하위 폴더와 같이 Configuration 폴더에서 이 서버 모델에 사용되는 폴더의 이름을 가져옵니다.

인수

없음

반환값

폴더의 이름을 나타내는 문자열을 반환합니다.

dom.serverModel.getServerIncludeUrlPatterns()

지원 버전

Dreamweaver MX

설명

다음 속성 목록을 반환하며, 이를 사용하면 다음에 액세스할 수 있습니다.

- 변환기 URL 패턴
- 파일 참조
- 유형

인수

없음

반환값

searchPattern마다 객체의 목록을 하나씩 반환합니다. 각 객체에는 다음 세 가지 속성이 있습니다.

속성	설명
pattern	EDML 파일의 searchPattern 필드에 지정된 JavaScript 정규식. 정규식은 슬래시 두 개 (//)로 구분됩니다.
fileRef	포함된 파일 참조에 해당하는 정규식 하위 일치 항목의 1부터 시작하는 인덱스
type	_includeUrl 접미사를 제거한 후 남는 paramName 값의 일부. 이 형식은 <MM:BeginLock> 태그의 type 속성에 할당됩니다. 예제를 보려면 Configuration/Translators 폴더의 Server Model SSI.htm을 참조하십시오.

예제

참여자 파일의 다음 코드 단편에서는 변환기 searchPatterns 태그를 보여 줍니다.

```
<searchPatterns whereToSearch="comment">
  <searchPattern paramNames=", ssi_comment_includeUrl">
    <![CDATA[<!--\s*#include\s+(file|virtual)\s*=\s*"([^\"]*)" \s*-->/i]]>
  </searchPattern>
</searchPatterns>
```

검색 패턴에는 두 하위 일치 항목(둘 다 괄호 안에 포함)을 지정하는 JavaScript 정규식이 포함됩니다. 첫 번째 하위 일치 항목은 텍스트 문자열 file 또는 virtual에 대한 것이고, 두 번째 하위 일치 항목은 파일 참조입니다.

변환기 URL 패턴에 액세스하려면 사용자의 코드가 다음 예제와 비슷해야 합니다.

```
var serverModel = dw.getDocumentDOM().serverModel;
var includeArray = new Array();
includeArray = serverModel.getServerIncludeUrlPatterns();
```

serverModel.getServerIncludeUrlPatterns()를 호출하면 다음 세 속성이 반환됩니다.

속성	반환값
pattern	/<!--\s*#include\s+(file virtual)\s*=\s*"([^\"]*)" \s*-->/i
fileRef	2
type	ssi_comment

dom.serverModel.getServerInfo()

지원 버전

Dreamweaver MX

설명

현재 서버 모델에 대한 정보를 반환합니다. 이 정보는 Configuration/ServerModels 폴더에 있는 서버 모델의 HTML 정의 파일에 정의됩니다.

HTML 정의 파일에 있는 정보를 수정하거나 파일에 변수 값 또는 함수를 추가할 수 있습니다. 예를 들어, serverName, serverLanguage 및 serverVersion 속성을 수정할 수 있습니다. dom.serverModel.getServerInfo() 함수는 서버 모델 제작자가 정의 파일에 추가하는 정보를 반환합니다.

참고: 기본 서버 모델 파일에 정의된 다른 값은 내부에서만 사용됩니다.

serverName, serverLanguage 및 serverVersion 속성은 특별히 개발자가 다음과 같은 해당 함수를 사용하여 직접 액세스할 수 있습니다.

- dom.serverModel.getServerName()
- dom.serverModel.getServerLanguage()
- dom.serverModel.getServerVersion()

인수

없음

반환값

현재 서버 모델에 대한 다양한 정보를 포함하는 JavaScript 객체를 반환합니다.

dom.serverModel.getServerName()

지원 버전

Dreamweaver 1, Dreamweaver MX에서 기능이 향상됨

설명

문서와 연관된 서버 이름을 가져오고 해당 값을 반환합니다. 서버 이름으로 서버 기술(예: ASP.NET 및 JSP)을 구별할 수 있지만 같은 서버 기술의 언어(예: ASP.NET VB 및 ASP.NET C#)는 구별할 수 없습니다. 가능한 값은 ASP, ASP.NET, Cold Fusion, JSP, PHP 등입니다.

문서와 연관된 서버 모델 이름을 가져오려면 324페이지의 “[dom.serverModel.getDisplayName\(\)](#)” 또는 325페이지의 “[dom.serverModel.getFolderName\(\)](#)”을 참조하십시오.

참고: Dreamweaver MX 이상 버전의 경우 *dom.serverModel.getServerName()*은 서버 모델 API의 *getServerInfo()* 함수 호출로 반환되는 객체의 *serverName* 속성을 읽습니다.

인수

없음

반환값

서버 이름을 포함하는 문자열을 반환합니다.

dom.serverModel.getServerSupportsCharset()

지원 버전

Dreamweaver MX

설명

문서와 연관된 서버 모델이 해당 이름의 문자 세트를 지원하는지 여부를 확인합니다.

참고: Dreamweaver는 사용자가 JavaScript 레이어에서 이 함수를 호출하도록 하며 또한 사용자가 페이지의 속성 대화 상자에 서 인코딩을 변경할 때 이 함수를 호출합니다. 서버 모델이 새 문자 인코딩을 지원하지 않는 경우 이 함수는 *false*를 반환하며, 변환할 것인지 묻는 경고 대화 상자가 표시됩니다. 예를 들어, 사용자가 ColdFusion 4.5 문서를 UTF-8로 변환하려고 하는 경우 ColdFusion은 UTF-8 인코딩을 지원하지 않으므로 이 경고 대화 상자가 표시됩니다.

인수

metaCharSetString

- **metaCharSetString** 인수는 특정 문자 세트의 이름을 지정하는 문자열 값입니다. 이 값은 문서와 연관된 meta 태그의 "charset=" 속성 값과 동일합니다. 지정된 서버 모델이 지원하는 값은 Configuration/ServerModels 폴더에 있는 해당 서버 모델의 HTML 정의 파일에 정의됩니다.

반환값

부울 값을 반환합니다. 서버 모델이 해당 이름의 문자 세트를 지원하면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.serverModel.getServerVersion()

지원 버전

UltraDev 1, Dreamweaver MX에서 기능이 향상됨

설명

문서와 연관된 서버 모델을 확인하고 해당 값을 반환합니다. 각 서버 모델에는 서버 모델 API에 정의된 것처럼 이름-버전 쌍으로 구성된 표를 반환하는 `getVersionArray()` 함수가 포함되어 있습니다.

참고: Dreamweaver의 경우 `dom.serverModel.getServerVersion()`은 먼저 서버 모델 API의 `getServerInfo()`를 호출하여 반환되는 객체의 `serverVersion` 속성을 읽습니다. 이 속성이 존재하지 않는 경우 `dom.serverModel.getServerVersion()`은 `getVersionArray()` 함수에서 이 속성을 읽습니다.

인수

name

- **name** 인수는 서버 모델의 이름을 나타내는 문자열입니다.

반환값

해당 이름의 서버 모델 버전을 포함하는 문자열을 반환합니다.

dom.serverModel.testAppServer()

지원 버전

Dreamweaver MX

설명

응용 프로그램 서버에 연결할 수 있는지 여부를 테스트합니다.

인수

없음

반환값

응용 프로그램 서버에 대한 연결 요청이 성공적인지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.getServerModels()

지원 버전

Dreamweaver MX

설명

현재 정의되어 있는 모든 서버 모델의 이름을 가져옵니다. 이름 집합은 [사이트 정의] 대화 상자의 [서버 모델] 텍스트 필드에 나타나는 것과 동일합니다.

인수

없음

반환값

문자열 배열을 반환합니다. 각 문자열 요소에는 현재 정의되어 있는 서버 모델의 이름이 포함됩니다.

17장: 디자인

Adobe® Dreamweaver®의 디자인 함수는 문서 모양의 디자인과 관련된 작업을 수행합니다. 이러한 작업에는 다음이 포함됩니다.

- 지정한 CSS 스타일 적용
- 선택한 프레임을 수직 또는 수평으로 분할
- 선택한 레이어 또는 핫스팟 정렬
- 선택한 플러그인 항목 재생
- 레이아웃 셀 만들기
- 표 행 또는 열 조작

CSS 레이아웃 함수

CSS 함수는 CSS 스타일 적용, 제거, 만들기 및 삭제 작업을 처리합니다. `dreamweaver.cssRuleTracker` 객체의 메서드는 선택 관리자에 있는 [CSS 규칙 추적기] 패널의 선택 영역을 제어하거나 선택 영역에 대해 작용합니다. `dreamweaver.cssStylePalette` 객체의 메서드는 현재 문서가 아니라 [스타일] 패널의 선택 영역을 제어하거나 선택 영역에 대해 작용합니다.

`dom.applyLayout()`

지원 버전

Dreamweaver CS3

설명

문서에 CSS 기반 레이아웃을 적용합니다. 문서 본문은 비어 있어야 하며 문서는 레이아웃을 적용할 수 있는 페이지여야 합니다. 즉, 다음과 같은 페이지입니다.

- HTML, XHTML, ColdFusion, PHP 등과 같이 HTML 기반이면서 CSS, XML, JavaScript 등이 아닌 페이지
- 프레임셋이나 템플릿 인스턴스(템플릿 자체인 경우 제외)가 아닌 페이지

인수

`layout-index`, `CSS`, `cssFileName`, `preventOverwrite`

- `layout-index`는 사용할 레이아웃을 지정하는 인덱스로, 0부터 시작하는 정수입니다. 이 인덱스는 레이아웃 목록의 인덱스로, 해당 함수에서 `layoutNames` 및 `layoutDescriptions`를 반환하는 데 사용됩니다.
- `CSS`는 CSS 레이아웃을 배치할 위치를 지정합니다. 사용할 수 있는 값은 다음과 같습니다.
 - “embed” - 문서의 head 섹션에 CSS를 포함합니다.
 - “link” - `cssFileName`에 링크합니다.
 - “create_and_link” - `cssFileName`에 CSS를 쓰고 링크합니다.
 - “import” - `cssFileName`을 가져옵니다.
 - “create_and_import” - `cssFileName`에 CSS를 쓰고 가져옵니다.
- `cssFileName`은 필요한 경우 링크하거나 가져와서 만들 CSS 파일 이름입니다.

- **preventOverwrite**는 다음과 같은 부울 값입니다. **true**: 새로운 CSS 파일을 만들 때 그 파일이 이미 존재하면 실패합니다. **false**: 파일이 이미 존재하는 경우 해당 파일을 덮어씁니다.

반환값

부울 값을 반환합니다. **true**: 레이아웃을 적용했습니다. **false**:를 반환합니다.

예제

```
dw.getLayoutNames();
var theDOM = dw.getDocumentDOM();
alert (theDOM.canApplyLayout());
if (theDOM.canApplyLayout())
    theDOM.applyLayout(1, "embed");
else
    alert("can't apply layout to this doc");
```

dom.canApplyLayout()

지원 버전

Dreamweaver CS3

설명

문서에 CSS 기반 레이아웃을 적용할 수 있는지 여부를 확인합니다. 문서 본문이 비어 있고 레이아웃을 적용할 수 있는 페이지인지 확인합니다. 즉, 다음과 같은 페이지입니다.

- HTML, XHTML, ColdFusion, PHP 등과 같이 기본적으로 HTML 기반이면서 CSS, XML, JavaScript 등이 아닌 페이지
- 프레임셋나 템플릿 인스턴스(템플릿 자체인 경우 제외)가 아닌 페이지

인수

없음

반환값

부울 값, 레이아웃을 적용할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dw.GetFilesForLayout()

지원 버전

Dreamweaver CS3

설명

지정한 레이아웃의 구성 파일 경로를 가져옵니다.

인수

layoutIndex

- **layoutIndex**는 레이아웃을 지정하는 인덱스로, 0부터 시작하는 정수입니다. 이 인덱스는 레이아웃 목록의 인덱스로, 해당 함수에서 **layoutNames** 및 **layoutDescriptions**를 반환하는 데 사용됩니다.

반환값

HTML과 미리 보기 이미지 파일의 전체 경로를 포함하는 문자열 배열을 반환합니다(null일 수 있음).

dw.getLayoutNames()

지원 버전

Dreamweaver CS3

설명

사용 가능한 CSS 기반 레이아웃의 이름을 가져옵니다.

인수

없음

반환값

레이아웃 이름의 문자열 배열을 반환합니다.

dw.getLayoutDescriptions()

지원 버전

Dreamweaver CS3

설명

사용 가능한 CSS 기반 레이아웃에 대한 설명을 가져옵니다.

인수

없음

반환값

레이아웃 설명의 문자열 배열을 반환합니다.

dom.applyCSSStyle()

지원 버전

Dreamweaver 4

설명

지정된 요소에 지정된 스타일을 적용합니다. 이 함수는 활성 문서에만 사용할 수 있습니다.

인수

elementNode, **styleName**, {**classOrID**}, {**bForceNesting**}

- **elementNode** 인수는 DOM의 요소 노드입니다. **elementNode** 인수가 null 값 또는 빈 문자열("")이면 함수는 현재 선택 영역에 대해 작동합니다.
- **styleName** 인수는 CSS 스타일의 이름입니다.
- 선택적인 **classOrID** 인수는 스타일이 적용될 속성("class" 또는 "id")입니다. **elementNode** 인수가 null 값 또는 빈 문자열이고 선택 영역이 정확하게 태그 안에 놓이지 않은 경우 스타일은 SPAN 태그를 사용하여 적용됩니다. 선택 영역이 삽입점인 경우 추론 기법을 사용하여 스타일이 적용될 태그가 확인됩니다.

- 선택적인 **bForceNesting** 인수는 중첩이 가능한지 여부를 나타내는 부울 값입니다. **bForceNesting** 플래그가 지정되면 문서의 기존 태그가 수정되지 않고 SPAN 태그가 새로 삽입됩니다. 이 인수를 지정하지 않으면 기본값은 **false**입니다.

반환값

없음

예제

다음 코드에서는 SPAN 태그 안에 선택 영역을 넣거나 선택 영역 주위의 태그에 CLASS 속성을 적용하여 선택 영역에 red 스타일을 적용합니다.

```
var theDOM = dreamweaver.getDocumentDOM('document');
theDOM.applyCSSStyle('', 'red');
```

dom.getElementView()

지원 버전

Dreamweaver 8

설명

이 함수는 문서에서 현재 선택한 요소의 요소 뷰를 가져옵니다. 현재 선택한 요소가 **normal**인 경우 getElementView() 함수는 선택한 요소에서 **full** 또는 **hidden**인 상위 요소를 찾습니다.

인수

없음

반환값

선택한 요소의 상태를 나타내는 문자열로, 다음과 같은 값을 포함할 수 있습니다.

- "hidden" - [디자인] 뷰에서 내용의 일부나 전체를 숨길 수 있도록 하는 CSS 속성이 해당 요소에 있는지를 나타냅니다. 지원되는 CSS 속성은 다음과 같습니다.
 - overflow: hidden, scroll 또는 auto
 - display: none
- "full" - 해당 요소가 기본적으로 "hidden"이지만 setElementView("full") 함수가 설정한 대로 현재 "full" 뷰에 있음을 나타냅니다.
- "normal" - 해당 요소가 "hidden"도 아니고 "full"도 아님을 나타냅니다.

예제

다음 예제에서는 "hidden" 상태인 선택한 요소의 상태를 "full"로 변경합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM && getElementView() == "hidden"){
    currentDOM.setElementView("full");
}
```

dom.getShowDivBackgrounds()

지원 버전

Dreamweaver 8

설명

이 함수는 [레이아웃 블록 배경] 시각 도구의 상태를 가져옵니다.

인수

없음

반환값

부울 값, [레이아웃 블록 배경] 시각 도구가 설정되어 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

다음 예제에서는 [레이아웃 블록 배경] 시각 도구가 설정되어 있는지 확인한 다음 설정되어 있지 않으면 설정합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.getShowDivBackgrounds() == false){
    currentDOM.setShowDivBackgrounds(true);
}
```

dom.getShowDivBoxModel()

지원 버전

Dreamweaver 8

설명

이 함수는 [레이아웃 블록 상자 모델] 시각 도구의 상태를 가져옵니다.

인수

없음

반환값

부울 값, [레이아웃 블록 상자 모델] 시각 도구가 설정되어 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

다음 예제에서는 [레이아웃 블록 상자 모델] 시각 도구가 설정되어 있는지 확인한 다음 설정되어 있지 않으면 설정합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.getShowDivBoxModel() == false){
    currentDOM.setShowDivBoxModel(true);
}
```

dom.getShowDivOutlines()

지원 버전

Dreamweaver 8

설명

이 함수는 [레이아웃 블록 외곽선] 시각 도구의 상태를 가져옵니다.

인수

없음

반환값

부울 값, [레이아웃 블록 외곽선] 시각 도구가 설정되어 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

다음 예제에서는 [레이아웃 블록 외곽선] 시각 도구가 설정되어 있는지 확인한 다음 설정되어 있지 않으면 설정합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.getShowDivOutlines() == false){
    currentDOM.setShowDivOutlines(true);
}
```

dom.removeCSSStyle()**지원 버전**

Dreamweaver 3

설명

지정된 요소에서 CLASS 또는 ID 속성을 제거하거나 지정된 요소 주위의 SPAN 태그를 제거합니다. 이 함수는 활성 문서에만 사용할 수 있습니다.

인수

elementNode, {**classOrID**}

- **elementNode** 인수는 DOM의 요소 노드입니다. **elementNode** 인수가 빈 문자열("")로 지정되면 함수는 현재 선택 영역에 대해 작용합니다.
- 선택적인 **classOrID** 인수는 제거할 속성("class" 또는 "id")입니다. **classOrID** 인수가 지정되지 않은 경우 기본값은 "class"입니다. **elementNode** 인수에 대해 정의된 CLASS 속성이 없으면 **elementNode** 인수 주위의 SPAN 태그가 제거됩니다.

반환값

없음

dom.resetAllElementViews()**지원 버전**

Dreamweaver 8

설명

이 함수는 내부에서 생성된 CSS를 모두 제거하여 모든 문서 요소의 요소 뷰를 원래 뷰로 재설정합니다.

인수

{**forceRefresh**}

- 선택적인 **forceRefresh** 인수는 제거할 내부 CSS가 없는 경우 전체 문서의 렌더링을 새로 고칠지 여부를 지정하는 부울 값입니다. 값이 **true**이면 새로 고침이 수행됩니다. 기본값은 **false**입니다.

반환값

없음

예제

다음 예제에서는 렌더링을 새로 고치지 않고 모든 문서 요소의 요소 뷰를 재설정합니다.

```
var currentDOM = dw.getDocumentDOM();
currentDOM.resetAllElementViews(false);
```

dom.setElementView()

지원 버전

Dreamweaver 8

설명

이 함수는 문서에서 현재 선택한 요소의 요소 뷰를 설정합니다. 현재 선택한 요소가 "normal"인 경우 setElementView() 함수는 현재 선택한 요소에서 "full" 또는 "hidden"인 상위 요소를 찾습니다.

인수**view**

- 필수적인 **view** 인수는 현재 선택한 요소를 "full" 또는 "hidden"으로 설정하는 문자열입니다. 현재 선택한 요소가 "normal"인 경우 setElementView() 함수는 현재 선택한 요소에서 "full" 또는 "hidden"인 상위 요소를 찾습니다. 자세한 내용은 332페이지의 "[dom.getElementView\(\)](#)"를 참조하십시오. 사용할 수 있는 값은 다음과 같습니다.
 - "full" — 요소를 "full" 뷰에 배치하는 내부 CSS를 제거하여 요소가 원래 상태로 돌아가도록 합니다.
 - "hidden" — 현재 선택한 요소가 "hidden" 뷰에 있는 경우 모든 내용을 표시하는 CSS가 생성된 다음 해당 CSS가 내부 디자인 타임 스타일 시트로 적용됩니다.

반환값

없음

예제

332페이지의 "[dom.getElementView\(\)](#)"를 참조하십시오.

dom.setShowDivBackgrounds()

지원 버전

Dreamweaver 8

설명

이 함수는 [레이아웃 블록 배경] 시각 도구를 설정하거나 해제합니다.

인수**show**

- 필수적인 **show** 인수는 [레이아웃 블록 배경] 시각 도구를 설정할지 여부를 지정하는 부울 값입니다. **show**를 true로 설정하면 [레이아웃 블록 배경] 시각 도구가 설정됩니다.

반환값

없음

예제

332페이지의 “[dom.getShowDivBackgrounds\(\)](#)”를 참조하십시오.

dom.setShowDivBoxModel()

지원 버전

Dreamweaver 8

설명

이 함수는 [레이아웃 블록 상자 모델] 시각 도구를 설정하거나 해제합니다.

인수

show

- 필수적인 **show** 인수는 [레이아웃 블록 상자 모델] 시각 도구의 설정 여부를 지정하는 부울 값입니다. **show**를 true로 설정하면 [레이아웃 블록 상자 모델] 시각 도구가 설정됩니다.

반환값

없음

예제

333페이지의 “[dom.getShowDivBoxModel\(\)](#)”을 참조하십시오.

dom.setShowDivOutlines()

지원 버전

Dreamweaver 8

설명

이 함수는 [레이아웃 블록 외곽선] 시각 도구를 설정하거나 해제합니다.

인수

show

- 필수적인 **show** 인수는 [레이아웃 블록 외곽선] 시각 도구의 설정 여부를 지정하는 부울 값입니다. **show**를 true로 설정하면 [레이아웃 블록 외곽선] 시각 도구가 설정됩니다.

반환값

없음

예제

333페이지의 “[dom.getShowDivOutlines\(\)](#)”를 참조하십시오.

dreamweaver.cssRuleTracker.editSelectedRule()

지원 버전

Dreamweaver MX 2004

설명

사용자가 규칙 추적기에 현재 선택되어 있는 규칙을 편집할 수 있도록 합니다. 이 함수는 선택된 규칙을 CSS 속성 격자에 표시하고 필요한 경우 속성 격자와 이를 포함하고 있는 플로터를 표시합니다.

인수

없음

반환값

없음

활성자

449페이지의 “[dreamweaver.cssRuleTracker.canEditSelectedRule\(\)](#)”을 참조하십시오.

dreamweaver.cssRuleTracker.newRule()

지원 버전

Dreamweaver MX 2004

설명

사용자가 새 규칙을 만들 수 있도록 [새 CSS 스타일] 대화 상자를 엽니다.

인수

없음

반환값

없음

dreamweaver.cssStylePalette.applySelectedStyle()

지원 버전

Dreamweaver MX

설명

[스타일] 패널의 선택 영역에 따라 현재 활성 문서나 첨부된 스타일 시트에 선택한 스타일을 적용합니다.

인수

{pane}

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

없음

활성자

450페이지의 “[dreamweaver.cssStylePalette.canApplySelectedStyle\(\)](#)”을 참조하십시오.

dreamweaver.cssStylePalette.attachStyleSheet()**지원 버전**

Dreamweaver 4

설명

[스타일] 패널의 선택 영역에 따라 사용자가 현재 활성 문서나 첨부된 스타일 시트 중 하나에 스타일 시트를 첨부할 수 있는 대화 상자를 표시합니다.

인수

없음

반환값

없음

dreamweaver.cssStylePalette.deleteSelectedStyle()**지원 버전**

Dreamweaver 3

설명

[스타일] 패널에서 현재 선택한 스타일을 문서에서 삭제합니다.

인수

{pane}

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

없음

활성자

450페이지의 “[dreamweaver.cssStylePalette.canDeleteSelectedStyle\(\)](#)”을 참조하십시오.

dreamweaver.cssStylePalette.duplicateSelectedStyle()**지원 버전**

Dreamweaver 3

설명

[스타일] 패널에서 현재 선택한 스타일을 복제하고 [스타일 복제] 대화 상자를 표시하여 사용자가 새 스타일에 이름이나 선택기를 할당하도록 합니다.

인수

{pane}

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서의 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

없음

활성자

451 페이지의 “[dreamweaver.cssStylePalette.canDuplicateSelectedStyle\(\)](#)”을 참조하십시오.

dreamweaver.cssStylePalette.editSelectedStyle()

지원 버전

Dreamweaver 3

설명

[스타일] 패널에서 현재 선택한 스타일에 대한 [스타일 정의] 대화 상자를 엽니다.

인수

{pane}

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서의 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

없음

활성자

451 페이지의 “[dreamweaver.cssStylePalette.canEditSelectedStyle\(\)](#)”을 참조하십시오.

dreamweaver.cssStylePalette.editSelectedStyleInCodeview()

지원 버전

Dreamweaver 8

설명

이 함수는 [코드] 뷰로 전환한 다음 [스타일] 패널에서 현재 선택한 스타일에 대한 코드로 마우스 포인터를 이동합니다.

인수**{pane}**

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

없음

활성자451페이지의 “[dreamweaver.cssStylePalette.canEditSelectedStyleInCodeview\(\)](#)”를 참조하십시오.**dreamweaver.cssStylePalette.editStyleSheet()****지원 버전**

Dreamweaver 3

설명

[스타일 시트 편집] 대화 상자를 엽니다.

인수

없음

반환값

없음

활성자452페이지의 “[dreamweaver.cssStylePalette.canEditStyleSheet\(\)](#)”를 참조하십시오.**dreamweaver.cssStylePalette.getDisplayStyles()****지원 버전**

Dreamweaver 8

설명

이 함수는 CSS 스타일을 렌더링할지 여부를 결정합니다. 기본값은 true입니다.

인수

없음

반환값

부울 값을 반환합니다. CSS 스타일이 렌더링되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

```
var areStylesRendered = dw.cssStylePalette.getDisplayStyles();
```

dreamweaver.cssStylePalette.getMediaType()

지원 버전

Dreamweaver MX 2004

설명

렌더링에 사용할 대상 미디어 유형을 가져옵니다. 기본 미디어 유형은 "screen"입니다.

인수

없음

반환값

대상 미디어 유형을 지정하는 문자열 값을 반환합니다.

예제

```
var mediaType = dw.cssStylePalette.getMediaType();
```

dreamweaver.cssStylePalette.getSelectedStyle()

지원 버전

Dreamweaver 3, fullSelector는 Dreamweaver MX에서 지원

설명

[스타일] 패널에서 현재 선택한 스타일의 이름을 가져옵니다.

인수

fullSelector

- fullSelector** 인수는 전체 선택기가 반환되는지, 아니면 클래스만 반환되는지를 나타내는 부울 값입니다. 아무 것도 지정하지 않으면 클래스 이름만 반환됩니다. 예를 들어, p.class1은 스타일이 class1의 모든 p 태그에 적용되지만 class1의 div 태그에는 적용되지 않음을 의미하는 선택기입니다. **fullSelector** 인수가 없으면 dreamweaver.cssStylePalette.getSelectedStyle() 함수는 선택기에 대한 클래스 이름 class1만 반환합니다. **fullSelector** 인수는 함수에서 class1 대신 p.class1을 반환하도록 설정합니다.

반환값

fullSelector 인수의 값이 true이면 함수는 전체 선택기를 반환하고, 스타일 시트 노드가 선택된 경우에는 빈 문자열을 반환합니다.

fullSelector 인수의 값이 false이거나 생략되면 선택한 스타일의 클래스 이름을 나타내는 문자열을 반환합니다. 선택한 스타일에 클래스가 없거나 스타일 시트 노드가 선택된 경우에는 빈 문자열을 반환합니다.

예제

스타일 red가 선택된 경우 dw.cssStylePalette.getSelectedStyle() 함수를 호출하면 "red"가 반환됩니다.

dreamweaver.cssStylePalette.getStyles()

지원 버전

Dreamweaver 3

설명

활성 문서에 있는 모든 클래스 스타일의 목록을 가져옵니다. 인수가 없으면 클래스 선택기 이름만 반환합니다. **bGetIDs** 인수가 true이면 ID 선택기 이름만 반환합니다. 두 경우 모두, **bGetFullSelector** 인수가 true이면 전체 선택기 이름을 반환합니다.

예를 들어, HTML 파일에 다음 코드가 있습니다.

```
<style>
.test{ background:none };
p.foo{ background:none };
#bar {background:none };
div#hello p.world {background:none};
```

다음 표의 호출은 [결과] 열에 값을 반환합니다.

함수 호출	결과
dw.cssStylePalette.getStyles()	foo, test, world
dw.cssStylePalette.getStyles(true)	bar,hello
dw.cssStylePalette.getStyles(false, true)	p.foo, .test, div#hello p.world
dw.cssStylePalette.getStyles(true, true)	#bar, div#hello p.world

인수

{bGetIDs}, {bGetFullSelector}

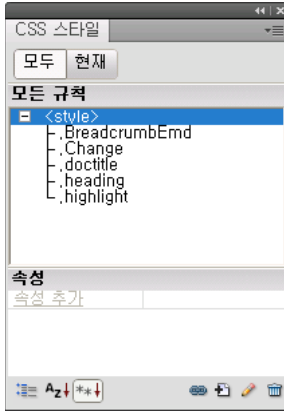
- **bGetIDs** 인수는 선택 사항입니다. 부울 값이며 true인 경우 함수는 ID 선택기 이름("#" 다음 부분)만 반환합니다. 기본값은 false입니다.
- **bGetFullSelector** 인수는 선택 사항입니다. 부울 값이며 true인 경우 이름 대신 전체 선택기 문자열을 반환합니다. 기본값은 false입니다.

반환값

문서에 있는 모든 클래스 스타일의 이름을 나타내는 문자열의 배열을 반환합니다.

예제

다음 그림과 같이 [스타일] 패널이 설정된 경우 `dreamweaver.cssStylePalette.getStyles()` 함수를 호출하면 "BreadcrumbEnd", "change", "doctitle", "heading" 및 "highlight" 문자열이 포함된 배열이 반환됩니다.



dreamweaver.cssStylePalette.newStyle()

지원 버전

Dreamweaver 3

설명

[새 스타일] 대화 상자를 엽니다.

인수

없음

반환값

없음

dreamweaver.cssStylePalette.renameSelectedStyle()

지원 버전

Dreamweaver 3

설명

[스타일] 패널에서 현재 선택한 규칙에 사용되는 클래스 이름 및 선택한 규칙에 있는 클래스 이름의 모든 인스턴스 이름을 변경합니다.

인수

{pane}

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서의 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

없음

활성자452페이지의 “[dreamweaver.cssStylePalette.canRenameSelectedStyle\(\)](#)”을 참조하십시오.

dreamweaver.cssStylePalette.setDisplayStyles()

지원 버전

Dreamweaver 8

설명

이 함수는 CSS 스타일을 렌더링할지 여부를 결정하고 열려 있는 모든 문서의 렌더링을 새로 고칩니다.

인수**display**

- **display** 인수는 부울 값이며, CSS 스타일을 렌더링하려면 **true**로 설정하고 그렇지 않으면 **false**로 설정합니다.

반환값

없음

예제

다음 예제에서는 CSS 스타일에 대한 렌더링을 해제합니다.

```
dw.cssStylePalette.setDisplayStyles(false);
```

dreamweaver.cssStylePalette.setMediaType()

지원 버전

Dreamweaver MX 2004

설명

렌더링에 사용할 대상 미디어 유형을 설정합니다. 열려 있는 모든 문서의 렌더링을 새로 고칩니다.

인수**mediaType**

- **mediaType** 인수는 새 대상 미디어 유형을 지정합니다.

반환값

없음

예제

```
dw.cssStylePalette.setMediaType("print");
```

dreamweaver.getBlockVisBoxModelColors()

지원 버전

Dreamweaver 8

설명

이 함수는 [레이아웃 블록 상자 모델] 시각 도구가 설정되어 있는 경우 선택한 블록에 대한 상자 모델을 렌더링하는 데 사용되는 색상을 가져옵니다.

인수

없음

반환값

다음 두 문자열을 포함하는 문자열 배열을 반환합니다.

- `marginColor` - #RRGGBB 형식의 RGB 색상에 대한 16진수 값입니다.
- `paddingColor` - #RRGGBB 형식의 RGB 색상에 대한 16진수 값입니다.

예제

다음 예제에서는 여백 및 패딩 색상의 값을 확인하고 흰색이 아닌 경우 둘 다 흰색으로 설정합니다.

```
var boxColors = dreamweaver.getBlockVisBoxModelColors();
if ((boxColors[0] != "#FFFFFF") || (boxColors[1] != "#FFFFFF")){
    currentDOM.setBlockVisBoxModelColors("#FFFFFF", "#FFFFFF");
}
```

dreamweaver.getBlockVisOutlineProperties()

지원 버전

Dreamweaver 8

설명

이 함수는 블록 시각화 시각 도구에 대한 외곽선 속성을 가져옵니다.

인수

forWhat

- 필수적인 **forWhat** 인수는 문자열입니다. 사용할 수 있는 값은 "divs", "selectedDiv" 또는 "layers"입니다. **forWhat** 인수가 "divs"인 경우 함수는 모든 레이아웃 블록의 외곽선을 지정하는 시각 도구에 사용된 속성을 반환합니다. **forWhat** 인수가 "selectedDiv"인 경우 함수는 선택한 레이아웃 블록의 외곽선을 지정하는 시각 도구에 사용된 속성을 반환합니다. **layers** 값은 레이어를 지정합니다.

반환값

다음 세 문자열을 포함하는 문자열 배열입니다.

- `color` - #RRGGBB 형식의 RGB 색상에 대한 16진수 값입니다.
- `width` - 픽셀 단위의 폭을 나타냅니다.
- `style` - "SOLID", "DOTTED", "DASHED" 또는 "OUTSET"입니다.

예제

다음 예제에서는 "divs"에 대한 외곽선 속성을 가져온 다음 외곽선 스타일을 "SOLID"로 설정합니다.

```
var outlineStyle = dw.getBlockVisOutlineProperties("divs");
if (outlineStyle[2] != "SOLID"){
    dw.setBlockVisOutlineProperties("divs", outlineStyle[0], outlineStyle[1], "SOLID");
}
```

dreamweaver.getDivBackgroundColors()

지원 버전

Dreamweaver 8

설명

이 함수는 [레이아웃 블록 배경] 시각 도구에 사용되는 색상을 가져옵니다.

인수

없음

반환값

16가지 색상을 포함하는 문자열 배열로서, 각 색상은 #RRGGBB 형식의 RGB 색상에 대한 16진수 값으로 표시됩니다.

예제

다음 예제에서는 [레이아웃 블록 배경] 시각 도구에 사용되는 배경색을 가져옵니다.

```
var backgroundColors = dreamweaver.getDivBackgroundColors();
```

dreamweaver.setBlockVisOutlineProperties()

지원 버전

Dreamweaver 8

설명

이 함수는 블록 시각화 시각 도구에 대한 외곽선 속성을 설정합니다.

인수

forWhat, color, width, {style}

- 필수적인 **forWhat** 인수는 지정한 색상과 폭이 적용될 대상을 지정하는 문자열입니다. 사용할 수 있는 값은 "divs", "selectedDiv" 또는 "layers"입니다. 값이 "layers"인 경우 [레이아웃 블록 외곽선] 시각 도구가 설정되어 있으면 지정한 색상과 폭이 모든 레이어의 외곽선을 지정하는 데 사용됩니다. 값이 "divs"인 경우 **color** 및 **width** 인수가 모든 div 및 다른 레이아웃 블록의 외곽선을 지정하는 데 사용됩니다. 값이 "selectedDiv"인 경우 **color** 및 **width** 인수가 선택한 div 또는 레이아웃 블록의 외곽선을 지정하는 데 사용됩니다.
- 필수적인 **color** 인수는 #RRGGBB 형식의 RGB 색상을 나타내는 16진수 값을 포함하는 문자열입니다.
- 필수적인 **width** 인수는 외곽선 폭을 픽셀 단위로 나타내는 정수입니다.
- 선택적인 **style** 인수는 외곽선의 스타일을 나타내는 문자열입니다. 사용할 수 있는 값은 "SOLID", "DOTTED", "DASHED" 및 "OUTSET"입니다. "OUTSET" 값은 레이어에만 적용할 수 있습니다. **forWhat** 인수의 값이 "selectedDiv"인 경우 이 인수는 무시됩니다.

반환값

없음

예제

345페이지의 “[dreamweaver.getBlockVisOutlineProperties\(\)](#)”를 참조하십시오.

dreamweaver.setDivBackgroundColors()**지원 버전**

Dreamweaver 8

설명

이 함수는 [레이아웃 블록 배경] 시각 도구에 사용되는 색상을 설정합니다.

인수**colors**

- 필수적인 **colors** 인수는 #RRGGBB 형식의 16진수 값으로 표시되는 모든 색상을 포함하는 문자열 배열입니다. 배열은 16개의 색상을 포함해야 합니다.

반환값

없음

예제

다음 예제에서는 div 배경색으로 16개의 색상만 지정되었는지 확인하고 그럴 경우 배경색으로 사용되는 색상을 회색 음영으로 설정합니다.

```
var currentDOM = dw.getDocumentDOM();
var divColors = currentDOM.getDivBackgroundColors("divs");
var shadesOfGray = new Array["#000000", "#111111", "#222222", "#333333", "#444444", "#555555", "#666666", "#777777", "#888888", "#999999", "#AAAAAA", "#BBBBBB", "#CCCCCC", "#DDDDDD", "#EEEEEE", "#FFFFFF"];
var howManyColors = divColors.length;
if (howManyColors <= 16) {
    for (var i = 0; i < howManyColors; i++) {
        currentDOM.setDivBackgroundColors("divs", shadesOfGray[i]);
    }
}
```

프레임 및 프레임세트 함수

프레임 및 프레임세트 함수는 프레임세트의 프레임 이름을 가져오거나 하나의 프레임을 두 개로 분할합니다.

dom.getFrameNames()**지원 버전**

Dreamweaver 3

설명

프레임세트에서 이름이 있는 모든 프레임의 목록을 가져옵니다.

인수

없음

반환값

각 문자열이 현재 프레임세트에 있는 프레임 이름을 나타내는 문자열의 배열을 반환합니다. 이름이 없는 프레임은 건너뛩니다. 프레임세트에 있는 모든 프레임에 이름이 없으면 빈 배열이 반환됩니다.

예제

이름이 지정된 두 프레임을 포함하여 모두 네 개의 프레임이 있는 문서에서 `dom.getFrameNames()` 함수를 호출하면 다음 문자열이 포함된 배열이 반환될 수 있습니다.

- "navframe"
- "main_content"

dom.isDocumentInFrame()

지원 버전

Dreamweaver 4

설명

현재 문서가 프레임세트 내에서 표시될지 여부를 식별합니다.

인수

없음

반환값

부울 값을 반환합니다. 문서가 프레임세트 안에 있으면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

dom.saveAllFrames()

지원 버전

Dreamweaver 4

설명

문서가 프레임세트이거나 프레임세트 안에 있는 경우 이 함수는 [문서] 윈도우의 모든 프레임과 프레임세트를 저장합니다. 지정된 문서가 프레임세트 안에 없는 경우 이 함수는 해당 문서를 저장합니다. 이전에 저장되지 않은 문서의 경우에는 [다른 이름으로 저장] 대화 상자를 엽니다.

인수

없음

반환값

없음

dom.splitFrame()

지원 버전

Dreamweaver 3

설명

선택한 프레임을 수직 또는 수평으로 분할합니다.

인수

splitDirection

- **splitDirection** 인수는 "up", "down", "left" 또는 "right" 방향 중 하나를 지정해야 하는 문자열입니다.

반환값

없음

활성자

440페이지의 “[dom.canSplitFrame\(\)](#)”을 참조하십시오.

레이어 및 이미지 맵 함수

레이어 및 이미지 맵 함수는 레이어와 이미지 맵 핫스팟의 정렬, 크기 조정 및 이동을 처리합니다. 함수 설명은 함수가 레이어에 적용되는지, 아니면 핫스팟에 적용되는지를 나타냅니다.

dom.align()

지원 버전

Dreamweaver 3

설명

선택된 레이어나 핫스팟을 왼쪽, 오른쪽, 위쪽 또는 아래쪽으로 정렬합니다.

인수

alignDirection

- **alignDirection** 인수는 레이어나 핫스팟의 정렬 기준이 되는 가장자리를 지정하는 문자열("left", "right", "top" 또는 "bottom")입니다.

반환값

없음

활성자

431페이지의 “[dom.canAlign\(\)](#)”을 참조하십시오.

dom.arrange()

지원 버전

Dreamweaver 3

설명

선택된 핫스팟을 지정한 방향으로 이동합니다.

인수

toBackOrFront

- **toBackOrFront** 인수는 핫스팟이 이동할 방향으로, **front** 또는 **back**입니다.

반환값

없음

활성자

432페이지의 “[dom.canArrange\(\)](#)”를 참조하십시오.

dom.makeSizesEqual()

지원 버전

Dreamweaver 3

설명

선택된 레이어나 핫스팟의 높이나 폭 또는 높이, 폭 모두를 동일하게 합니다. 마지막으로 선택된 레이어나 핫스팟이 기준입니다.

인수

bHoriz, bVert

- **bHoriz** 인수는 레이어나 핫스팟의 가로 크기를 조정할지 여부를 나타내는 부울 값입니다.
- **bVert** 인수는 레이어나 핫스팟의 세로 크기를 조정할지 여부를 나타내는 부울 값입니다.

반환값

없음

dom.moveSelectionBy()

지원 버전

Dreamweaver 3

설명

선택된 레이어나 핫스팟을 지정한 픽셀 수만큼 가로 또는 세로로 이동합니다.

인수

x, y

- **x** 인수는 선택 영역을 가로로 이동할 픽셀 수입니다.
- **y** 인수는 선택 영역을 세로로 이동할 픽셀 수입니다.

반환값

없음

dom.resizeSelectionBy()

지원 버전

Dreamweaver 3

설명

현재 선택된 레이어나 핫스팟의 크기를 조정합니다.

인수

left, top, bottom, right

- **left** 인수는 레이어나 핫스팟의 왼쪽 경계선에 대한 새 위치입니다.
- **top** 인수는 레이어나 핫스팟의 위쪽 경계선에 대한 새 위치입니다.
- **bottom** 인수는 레이어나 핫스팟의 아래쪽 경계선에 대한 새 위치입니다.
- **right** 인수는 레이어나 핫스팟의 오른쪽 경계선에 대한 새 위치입니다.

반환값

없음

예제

선택된 레이어의 왼쪽, 위쪽, 폭 및 높이 속성이 다음과 같은 경우 `dw.getDocumentDOM().resizeSelectionBy(10,30,30,10)`를 호출하면 왼쪽 40, 위쪽 20, 폭 240, 높이 240으로 다시 설정하는 것과 동일한 결과가 나타납니다.

dom.setLayerTag()

지원 버전

Dreamweaver 3

설명

선택된 레이어를 정의하는 HTML 태그를 지정합니다.

인수

tagName

- **tagName** 인수는 "layer", "ilayer", "div" 또는 "span"이어야 합니다.

반환값

없음

레이아웃 환경 함수

레이아웃 환경 함수는 문서에 적용되는 설정과 관련된 작업을 처리합니다. 바탕 이미지의 소스, 위치 및 불투명도에 적용되고, 눈금자 원점 및 단위를 가져오고 설정하며, 격자를 활성화하거나 비활성화하고, 설정을 변경하고, 플러그인의 재생을 시작하거나 중단합니다.

dom.getRulerOrigin()

지원 버전

Dreamweaver 3

설명

눈금자의 원점을 가져옵니다.

인수

없음

반환값

두 정수로 구성된 배열을 반환합니다. 첫 번째 배열 항목은 원점의 **x** 좌표이고, 두 번째 배열 항목은 원점의 **y** 좌표입니다. 두 값 모두 픽셀 단위입니다.

dom.getRulerUnits()

지원 버전

Dreamweaver 3

설명

현재의 눈금자 단위를 가져옵니다.

인수

없음

반환값

다음 값 중 하나가 들어 있는 문자열을 반환합니다.

- "in"
- "cm"
- "px"

dom.getTracingImageOpacity()

지원 버전

Dreamweaver 3

설명

문서의 바탕 이미지에 대한 불투명도 설정을 가져옵니다.

인수

없음

반환값

0부터 100 사이의 값을 반환하고, 불투명도가 설정되어 있지 않은 경우 아무 것도 반환하지 않습니다.

활성자

441 페이지의 “[dom.hasTracingImage\(\)](#)”를 참조하십시오.

dom.loadTracingImage()

지원 버전

Dreamweaver 3

설명

[이미지 소스 선택] 대화 상자를 엽니다. 사용자가 이미지를 선택하고 [확인]을 클릭하면 [바탕 이미지] 필드의 내용이 입력되어 있는 [페이지 속성] 대화 상자가 열립니다.

인수

없음

반환값

없음

dom.playAllPlugins()

지원 버전

Dreamweaver 3

설명

문서의 모든 플러그인 내용을 재생합니다.

인수

없음

반환값

없음

dom.playPlugin()

지원 버전

Dreamweaver 3

설명

선택한 플러그인 항목을 재생합니다.

인수

없음

반환값

없음

활성자

438페이지의 “[dom.canPlayPlugin\(\)](#)”을 참조하십시오.

dom.setRulerOrigin()

지원 버전

Dreamweaver 3

설명

눈금자의 원점을 설정합니다.

인수

xCoordinate, yCoordinate

- **xCoordinate** 인수는 가로 축의 값을 픽셀 단위로 표시한 것입니다.
- **yCoordinate** 인수는 세로 축의 값을 픽셀 단위로 표시한 것입니다.

반환값

없음

dom.setRulerUnits()

지원 버전

Dreamweaver 3

설명

현재의 눈금자 단위를 설정합니다.

인수

units

- **units** 인수는 "px", "in" 또는 "cm"이어야 합니다.

반환값

없음

dom.setTracingImagePosition()

지원 버전

Dreamweaver 3

설명

바탕 이미지의 왼쪽 상단 모서리를 지정한 좌표로 이동합니다. 인수를 생략하면 [바탕 이미지 위치 조정] 대화 상자가 나타납니다.

인수

x, y

- **x** 인수는 가로 좌표를 지정하는 픽셀 수입니다.
- **y** 인수는 세로 좌표를 지정하는 픽셀 수입니다.

반환값

없음

활성자

441페이지의 “[dom.hasTracingImage\(\)](#)”를 참조하십시오.

dom.setTracingImageOpacity()

지원 버전

Dreamweaver 3

설명

바탕 이미지의 불투명도를 설정합니다.

인수

opacityPercentage

- **opacityPercentage** 인수는 0에서 100 사이의 숫자여야 합니다.

반환값

없음

활성자

441페이지의 “[dom.hasTracingImage\(\)](#)”를 참조하십시오.

예제

다음 코드에서는 바탕 이미지의 불투명도를 30%로 설정합니다.

```
dw.getDocumentDOM().setTracingOpacity('30');
```

dom.snapTracingImageToSelection()

지원 버전

Dreamweaver 3

설명

바탕 이미지의 왼쪽 상단 모서리를 현재 선택 영역의 왼쪽 상단 모서리에 맞춥니다.

인수

없음

반환값

없음

활성자

441페이지의 “[dom.hasTracingImage\(\)](#)”를 참조하십시오.

dom.stopAllPlugins()

지원 버전

Dreamweaver 3

설명

문서에서 현재 재생 중인 모든 플러그인 내용을 중단합니다.

인수

없음

반환값

없음

dom.stopPlugin()

지원 버전

Dreamweaver 3

설명

선택한 플러그인 항목을 중단합니다.

인수

없음

반환값

선택 영역이 플러그인을 통해 현재 재생되고 있는지 여부를 나타내는 부울 값을 반환합니다.

활성자

440페이지의 “[dom.canStopPlugin\(\)](#)”을 참조하십시오.

dreamweaver.arrangeFloatingPalettes()**지원 버전**

Dreamweaver 3

설명

표시된 부동 패널을 해당 패널의 기본 위치로 이동합니다.

인수

없음

반환값

없음

dreamweaver.showGridSettingsDialog()**지원 버전**

Dreamweaver 3

설명

[격자 설정] 대화 상자를 엽니다.

인수

없음

반환값

없음

레이아웃 뷰 함수

레이아웃 뷰 함수는 문서 내의 레이아웃 요소를 변경하는 작업을 처리합니다. 이러한 함수는 위치, 속성 및 모양을 비롯한 표, 열 및 셀의 설정에 영향을 줍니다.

dom.getClickedHeaderColumn()**지원 버전**

Dreamweaver 4

설명

사용자가 [레이아웃] 뷰에서 표의 머리글에 있는 메뉴 버튼을 클릭하여 표 머리글 메뉴를 열면 이 함수는 사용자가 클릭한 열의 인덱스를 반환합니다. 표 머리글 메뉴가 표시되지 않는 경우의 결과는 정의되어 있지 않습니다.

인수

없음

반환값

열의 인덱스를 나타내는 정수를 반환합니다.

dom.getShowLayoutTableTabs()

지원 버전

Dreamweaver 4

설명

현재 문서의 [레이아웃] 뷰에 레이아웃 표의 탭이 표시되는지 여부를 확인합니다.

인수

없음

반환값

현재 문서의 [레이아웃] 뷰에 레이아웃 표의 탭이 표시되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.getShowLayoutView()

지원 버전

Dreamweaver 4

설명

현재 문서의 뷰가 [레이아웃] 뷰인지, 아니면 [표준] 뷰인지를 확인합니다.

인수

없음

반환값

현재 문서가 [레이아웃] 뷰에 있으면 **true**를 반환하고 문서가 [표준] 뷰에 있으면 **false**를 반환합니다.

dom.getShowBlockBackgrounds()

지원 버전

Dreamweaver 8

설명

이 함수는 모든 블록이나 **div**에 대해 배경색이 적용되도록 하는 시각 도구의 상태를 가져옵니다.

인수**allblocks**

- 필수적인 **allblocks** 인수는 부울 값입니다. 값을 **true**로 설정하면 **div** 태그에만 적용되고, **false**로 설정하면 모든 블록 요소에 적용됩니다.

반환값

부울 값을 반환합니다. 배경색이 적용되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

다음 예제에서는 모든 블록에 대해 배경색이 적용되었는지 확인한 다음 적용되지 않았으면 모든 블록에 대해 배경색을 적용합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.getShowBlockBackgrounds(false) == false){
    currentDOM.setShowBlockBackgrounds(false);
}
```

dom.getShowBlockBorders()

지원 버전

Dreamweaver 8

설명

이 함수는 모든 블록이나 모든 **div**에 대해 테두리를 표시하는 시각 도구의 상태를 가져옵니다.

인수**allblocks**

- 필수적인 **allblocks** 인수는 부울 값입니다. 값을 **true**로 설정하면 **div** 태그에 대한 상태만 가져오고, **false**로 설정하면 모든 블록 요소에 대한 상태를 가져옵니다.

반환값

부울 값, 테두리가 표시되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

다음 예제에서는 블록 테두리 시각 도구가 설정되어 있는지 확인한 다음 설정되어 있지 않으면 설정합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.getShowBlockBorders(false) == false){
    currentDOM.setShowBlockBorders(true);
}
```

dom.getShowBlockIDs()

지원 버전

Dreamweaver 8

설명

이 함수는 모든 블록이나 **div**에 대해 ID와 클래스 정보를 표시하는 시각 도구의 상태를 가져옵니다.

인수**allblocks**

- 필수적인 **allblocks** 인수는 부울 값입니다. 값을 **true**로 설정하면 **div** 태그에 대한 ID와 클래스만 표시되고, **false**로 설정하면 모든 블록 요소에 대한 ID와 클래스가 표시됩니다.

반환값

부울 값을 반환합니다. 부울 값을 반환합니다. **true**일 경우 ID가 표시되고, **false**일 경우 ID가 표시되지 않습니다.

예제

다음 예제에서는 블록 ID가 표시되는지 확인한 다음 표시되지 않으면 표시합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.getShowBlockIDs(false) == false){
    currentDOM.setShowBlockIDs(true);
}
```

dom.getShowBoxModel()**지원 버전**

Dreamweaver 8

설명

이 함수는 선택한 블록의 전체 상자 모델에 색상을 지정하는 시각 도구를 설정하거나 해제합니다.

인수

없음

반환값

없음

예제

다음 예제에서는 선택한 상자의 전체 상자 모델에 색상이 표시되는지 확인하고 표시되지 않으면 색상을 지정합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.getShowBoxModel() == false){
    currentDOM.setShowBoxModel(true);
}
```

dom.setShowBlockBackgrounds()**지원 버전**

Dreamweaver 8

설명

이 함수는 모든 블록이나 모든 **div**에 대해 배경색이 적용되도록 하는 시각 도구를 설정하거나 해제합니다.

인수**allblocks**

- 필수적인 **allblocks** 인수는 부울 값입니다. 값을 **true**로 설정하면 **div** 태그에만 배경색이 적용되고, **false**로 설정하면 모든 블록 요소에 배경색이 적용됩니다.

반환값

없음

예제

358페이지의 “[dom.getShowBlockBackgrounds\(\)](#)”를 참조하십시오.

dom.setShowBlockBorders()

지원 버전

Dreamweaver 8

설명

이 함수는 모든 블록이나 모든 **div**에 대해 테두리를 표시하는 시각 도구를 설정하거나 해제합니다.

인수**allblocks**

- 필수적인 **allblocks** 인수는 부울 값입니다. 값을 **true**로 설정하면 테두리가 **div** 태그에만 적용되고, **false**로 설정하면 테두리가 모든 블록 요소에 적용됩니다.

반환값

없음

예제

359페이지의 “[dom.getShowBlockBorders\(\)](#)”를 참조하십시오.

dom.setShowBlockIDs()

지원 버전

Dreamweaver 8

설명

이 함수는 모든 블록이나 모든 **div**에 대해 ID와 클래스를 표시하는 시각 도구를 설정하거나 해제합니다.

인수**allblocks**

- 필수적인 **allblocks** 인수는 부울 값입니다. 값을 **true**로 설정하면 **div** 태그에 대한 ID와 클래스만 표시되고, **false**로 설정하면 모든 블록 요소에 대한 ID와 클래스가 표시됩니다.

반환값

없음

예제

359페이지의 “[dom.getShowBlockIDs\(\)](#)”를 참조하십시오.

dom.setShowBoxModel()

지원 버전

Dreamweaver 8

설명

이 함수는 선택한 블록의 전체 상자 모델에 색상을 지정하는 시각 도구의 상태를 설정합니다.

인수

없음

반환값

부울 값을 반환합니다. 상자 모델이 표시되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

360페이지의 “[dom.getShowBoxModel\(\)](#)”을 참조하십시오.

dom.setShowLayoutTableTabs()

지원 버전

Dreamweaver 4

설명

[레이아웃] 뷰에 있을 때 항상 레이아웃 표의 탭이 표시되도록 현재 문서를 설정합니다. 이 함수는 문서를 [레이아웃] 뷰로 전환하지 않습니다.

인수

bShow

- **bShow** 인수는 현재 문서가 [레이아웃] 뷰에 있을 때 레이아웃 표의 탭을 표시할지 여부를 나타냅니다. **bShow**가 **true**이면 탭이 표시되고, **bShow**가 **false**이면 탭이 표시되지 않습니다.

반환값

없음

dom.setShowLayoutView()

지원 버전

Dreamweaver 4

설명

bShow가 **true**인 경우 현재 문서를 [레이아웃] 뷰로 전환합니다.

인수**bShow**

- **bShow** 인수는 현재 문서를 [레이아웃] 뷰와 [표준] 뷰 사이에서 전환하는 부울 값입니다. **bShow**가 true이면 현재 문서가 [레이아웃] 뷰에 표시되고, **bShow**가 false이면 현재 문서가 [표준] 뷰에 표시됩니다.

반환값

없음

확대/축소 함수

확대/축소 함수는 [디자인] 뷰에서 확대하거나 축소합니다.

dreamweaver.activeViewScale()

지원 버전

Dreamweaver 8

설명

activeViewScale 속성은 변경 가능한 부동 소수점 값을 가져오거나 설정합니다. 값을 가져오면 [확대/축소] 콤보 상자에 표시되는 활성 뷰 비율을 100으로 나눈 값이 반환됩니다. 예를 들어, 100%는 1.0, 50%는 0.5 등으로 반환됩니다. 사용자가 값을 설정하면 [확대/축소] 콤보 상자에 값이 설정됩니다. 값의 범위는 0.06 - 64.00(6% - 6400%)입니다.

예제

다음 예제에서는 현재 뷰의 비율 값을 가져온 후 비율이 100% 이하인 경우 확대합니다.

```
if (canZoom() && dreamweaver.activeViewScale <= 1.0) {
    zoomIn();
}
```

다음 예제에서는 현재 뷰의 비율 값을 50%로 설정합니다.

```
dreamweaver.activeViewScale = 0.50;
```

dreamweaver.fitAll()

지원 버전

Dreamweaver 8

설명

이 함수는 [디자인] 뷰의 현재 보이는 부분에 맞도록 전체 문서를 확대하거나 축소합니다.

인수

없음

반환값

없음

활성자

449페이지의 “[dreamweaver.canZoom\(\)](#)”을 참조하십시오.

예제

```
if (canZoom()) {  
    fitAll();  
}
```

dreamweaver.fitSelection()

지원 버전

Dreamweaver 8

설명

이 함수는 [디자인] 뷰의 현재 보이는 부분에 맞도록 현재 선택 영역을 확대하거나 축소합니다.

인수

없음

반환값

없음

활성자

444페이지의 “[dreamweaver.canFitSelection\(\)](#)”을 참조하십시오.

예제

```
if (canFitSelection()) {  
    fitSelection();  
}
```

dreamweaver.fitWidth()

지원 버전

Dreamweaver 8

설명

이 함수는 [디자인] 뷰의 현재 보이는 부분에 맞도록 전체 문서의 폭을 확대하거나 축소합니다.

인수

없음

반환값

없음

활성자

449페이지의 “[dreamweaver.canZoom\(\)](#)”을 참조하십시오.

예제

```
if (canZoom()) {
    fitWidth();
}
```

dreamweaver.zoomIn()**지원 버전**

Dreamweaver 8

설명

이 함수는 현재 활성 [디자인] 뷰를 확대합니다. 확대/축소 레벨은 [확대/축소율] 메뉴의 다음 사전 설정 값입니다. 다음 사전 설정 값이 없는 경우 이 함수는 작동하지 않습니다.

인수

없음

반환값

없음

활성자

449페이지의 “[dreamweaver.canZoom\(\)](#)”을 참조하십시오.

예제

```
if (canZoom()) {
    zoomIn();
}
```

dreamweaver.zoomOut()**지원 버전**

Dreamweaver 8

설명

이 함수는 현재 활성 [디자인] 뷰를 축소합니다. 확대/축소 레벨은 [확대/축소율] 메뉴의 다음 사전 설정 값입니다. 다음 사전 설정 값이 없는 경우 이 함수는 작동하지 않습니다.

인수

없음

반환값

없음

활성자

449페이지의 “[dreamweaver.canZoom\(\)](#)”을 참조하십시오.

예제

```
if (canZoom()) {
    zoomOut();
}
```

안내선 함수 및 속성

안내선 함수 및 속성을 사용하여 HTML 페이지의 요소를 측정하고 레이아웃을 지정할 수 있도록 하는 안내선을 표시, 조작 및 삭제할 수 있습니다.

dom.clearGuides()

지원 버전

Dreamweaver 8

설명

이 함수는 문서에 있는 안내선을 모두 삭제할지 여부를 결정합니다.

인수

없음

반환값

없음

예제

다음 예제에서는 문서에 안내선이 하나 이상 있는 경우 문서의 안내선을 모두 삭제합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.hasGuides() == true) {
    currentDOM.clearGuides();
}
```

dom.createHorizontalGuide()

지원 버전

Dreamweaver 8

설명

이 함수는 문서의 현재 위치에 수평 안내선을 만듭니다.

인수

location

- location** 인수는 값과 단위가 사이에 공백 없이 한 문자열로 표시되는 안내선의 위치입니다. 사용할 수 있는 단위는 픽셀의 경우 "px"이고 백분율의 경우 "%"입니다. 예를 들어, 10픽셀을 지정하려면 `location = "10px"`로 지정하고, 50%를 지정하려면 `location = "50%"`로 지정하십시오.

반환값

없음

예제

다음 예제에서는 문서의 현재 위치에 수평 안내선을 만듭니다.

```
var currentDOM = dw.getDocumentDOM();
currentDOM.createHorizontalGuide("10px");
```

dom.createVerticalGuide()**지원 버전**

Dreamweaver 8

설명

이 함수는 문서의 현재 위치에 수직 안내선을 만듭니다.

인수**location**

- **location** 인수는 값과 단위가 사이에 공백 없이 한 문자열로 표시되는 안내선의 위치입니다. 사용할 수 있는 단위는 픽셀의 경우 "px"이고 백분율의 경우 "%"입니다. 예를 들어, 10픽셀을 지정하려면 location = "10px"로 지정하고, 50%를 지정하려면 location = "50%"로 지정하십시오.

반환값

없음

예제

다음 예제에서는 문서의 현재 위치에 수직 안내선을 만듭니다.

```
var currentDOM = dw.getDocumentDOM();
currentDOM.createVerticalGuide("10px");
```

dom.deleteHorizontalGuide()**지원 버전**

Dreamweaver 8

설명

이 함수는 지정된 위치에 있는 수평 안내선을 삭제합니다.

인수**location**

- **location** 인수는 테스트할 문서의 위치를 나타내는 문자열로, 값과 단위가 사이에 공백 없이 한 문자열로 표시됩니다. 사용할 수 있는 단위는 픽셀의 경우 "px"이고 백분율의 경우 "%"입니다. 예를 들어, 10픽셀을 지정하려면 location = "10px"로 지정하고, 50%를 지정하려면 location = "50%"로 지정하십시오.

반환값

없음

예제

다음 예제에서는 문서의 지정 한 위치에 있는 수평 안내선을 삭제합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.hasHorizontalGuide("10px") == true) {
    currentDOM.deleteHorizontalGuide("10px");
}
```

dom.deleteVerticalGuide()**지원 버전**

Dreamweaver 8

설명

이 함수는 지정 한 위치에 있는 수직 안내선을 삭제합니다.

인수**location**

- **location** 인수는 테스트할 문서의 위치를 나타내는 문자열로, 값과 단위가 사이에 공백 없이 한 문자열로 표시됩니다. 사용할 수 있는 단위는 픽셀의 경우 "px"이고 백분율의 경우 "%"입니다. 예를 들어, 10픽셀을 지정하려면 location = "10px"로 지정 하고, 50%를 지정하려면 location = "50%"로 지정하십시오.

반환값

없음

예제

다음 예제에서는 문서의 지정 한 위치에 있는 수직 안내선을 삭제합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.hasVerticalGuide("10px") == true) {
    currentDOM.deleteVerticalGuide("10px");
}
```

dom.guidesColor**지원 버전**

Dreamweaver 8

설명

이 변경 가능한 색상 속성은 문서의 안내선 색상을 결정합니다. 사용자는 이 속성을 설정하고 가져올 수 있습니다.

인수

없음

반환값

없음

예제

다음 예제에서는 안내선을 회색으로 표시합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.guidesColor != "#444444") {
    currentDOM.guidesColor = "#444444";
}
```

dom.guidesDistanceColor

지원 버전

Dreamweaver 8

설명

이 변경 가능한 색상 속성은 문서 안내선의 거리 피드백 색상을 결정합니다. 사용자는 이 속성을 설정하고 가져올 수 있습니다.

인수

없음

반환값

없음

예제

다음 예제에서는 안내선의 거리 피드백 색상을 회색으로 지정합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.guidesDistanceColor != "#CCCCCC") {
    currentDOM.guidesDistanceColor = "#CCCCCC";
}
```

dom.guidesLocked

지원 버전

Dreamweaver 8

설명

이 변경 가능한 부울 속성은 문서에서 안내선을 잠글지 여부를 결정합니다. 사용자는 이 속성을 설정하고 가져올 수 있습니다.

인수

없음

반환값

없음

예제

다음 예제에서는 안내선이 잠기지 않은 경우 안내선을 잠급니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.guidesLocked == false) {
    currentDOM.guidesLocked = true;
}
```

dom.guidesSnapToElements

지원 버전

Dreamweaver 8

설명

이 변경 가능한 부울 속성은 안내선을 문서의 요소에 스냅할지 여부를 결정합니다. 사용자는 이 속성을 설정하고 가져올 수 있습니다.

인수

없음

반환값

없음

예제

다음 예제에서는 문서의 안내선을 요소에 스냅합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.guidesSnapToElements == false) {
    currentDOM.guidesSnapToElements = true;
}
```

dom.guidesVisible

지원 버전

Dreamweaver 8

설명

이 변경 가능한 부울 속성은 문서에 안내선을 표시할지 여부를 결정합니다. 사용자는 이 속성을 설정하고 가져올 수 있습니다.

인수

없음

반환값

없음

예제

다음 예제에서는 안내선이 표시되지 않는 경우 표시되도록 안내선을 설정합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.guidesVisible == false) {
    currentDOM.guidesVisible = true;
}
```

dom.hasGuides()

지원 버전

Dreamweaver 8

설명

이 함수는 문서에 하나 이상의 안내선이 있는지 여부를 확인합니다. 사용자는 이 속성을 설정하고 가져올 수 있습니다.

인수

없음

반환값

없음

예제

다음 예제에서는 문서에 안내선이 하나 이상 있는 경우 문서의 안내선을 모두 삭제합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.hasGuides() == true) {
    currentDOM.clearGuides();
}
```

dom.hasHorizontalGuide()

지원 버전

Dreamweaver 8

설명

이 함수는 문서의 지정 한 위치에 수평 안내선이 있는지 여부를 확인합니다.

인수**location**

- **location** 인수는 테스트할 문서의 위치를 나타내는 문자열로, 값과 단위가 사이에 공백 없이 한 문자열로 표시됩니다. 사용할 수 있는 단위는 픽셀의 경우 "px"이고 백분율의 경우 "%"입니다. 예를 들어, 10픽셀을 지정하려면 `location = "10px"`로 지정하고, 50%를 지정하려면 `location = "50%"`로 지정하십시오.

반환값

부울 값을 반환합니다. 해당 위치에 수평 안내선이 있으면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

예제

다음 예제에서는 문서의 지정 한 위치에 수평 안내선이 있는 경우 문서의 안내선을 모두 삭제합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.hasHorizontalGuide("10px") == true) {
    currentDOM.clearGuides();
}
```

dom.hasVerticalGuide()

지원 버전

Dreamweaver 8

설명

이 함수는 문서의 현재 위치에 수직 안내선이 있는지 여부를 확인합니다.

인수**location**

- **location** 인수는 테스트할 문서의 위치를 나타내는 문자열로, 값과 단위가 사이에 공백 없이 한 문자열로 표시됩니다. 사용할 수 있는 단위는 픽셀의 경우 "px"이고 백분율의 경우 "%"입니다. 예를 들어, 10픽셀을 지정하려면 `location = "10px"`로 지정하고, 50%를 지정하려면 `location = "50%"`로 지정하십시오.

반환값

부울 값을 반환합니다. 해당 위치에 수직 안내선이 있으면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

예제

다음 예제에서는 문서의 지정된 위치에 수직 안내선이 있는 경우 문서의 안내선을 모두 삭제합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.hasVerticalGuide("10px") == true) {
    currentDOM.clearGuides();
}
```

dom.snapToGuides

지원 버전

Dreamweaver 8

설명

이 변경 가능한 부울 속성은 요소를 문서의 안내선에 스냅할지 여부를 결정합니다. 사용자는 이 속성을 설정하고 가져올 수 있습니다.

인수

없음

반환값

없음

예제

다음 예제에서는 문서의 요소를 안내선에 스냅합니다.

```
var currentDOM = dw.getDocumentDOM();
if (currentDOM.snapToGuides == false) {
    currentDOM.snapToGuides = true;
}
```

표 편집 함수

표 함수는 표 행 및 열을 추가 및 제거하고, 열 폭과 행 높이를 변경하고, 단위를 픽셀과 백분율 사이에서 전환하고, 기타 표준 표 편집 작업을 수행합니다.

dom.convertWidthsToPercent()

지원 버전

Dreamweaver 3

설명

이 함수는 현재 표의 모든 WIDTH 속성을 픽셀에서 백분율로 변환합니다.

인수

없음

반환값

없음

dom.convertWidthsToPixels()

지원 버전

Dreamweaver 4

설명

이 함수는 현재 표의 모든 WIDTH 속성을 백분율에서 픽셀로 변환합니다.

인수

없음

반환값

없음

dom.decreaseColspan()

지원 버전

Dreamweaver 3

설명

이 함수는 가로 셀을 한 개 나눕니다.

인수

없음

반환값

없음

활성자

434페이지의 “[dom.canDecreaseColspan\(\)](#)”을 참조하십시오.

dom.decreaseRowspan()

지원 버전

Dreamweaver 3

설명

이 함수는 세로 셀을 한 개 나눕니다.

인수

없음

반환값

없음

활성자

434페이지의 “[dom.canDecreaseRowspan\(\)](#)”을 참조하십시오.

dom.deleteTableColumn()

지원 버전

Dreamweaver 3

설명

이 함수는 선택한 표 열을 제거합니다.

인수

없음

반환값

없음

활성자

434페이지의 “[dom.canDeleteTableColumn\(\)](#)”을 참조하십시오.

dom.deleteTableRow()

지원 버전

Dreamweaver 3

설명

이 함수는 선택한 표 행을 제거합니다.

인수

없음

반환값

없음

활성자

435페이지의 “[dom.canDeleteTableRow\(\)](#)”를 참조하십시오.

dom.doDeferredTableUpdate()

지원 버전

Dreamweaver 3

설명

[일반] 환경 설정에서 [보다 빠른 표 편집] 옵션을 선택한 경우 이 함수는 표 외부를 선택하지 않더라도 표 레이아웃에 최근 변경 사항이 적용되게 합니다. [보다 빠른 표 편집] 옵션을 선택하지 않는 경우 이 함수는 아무 효과가 없습니다.

인수

없음

반환값

없음

dom.getShowTableWidths()

지원 버전

Dreamweaver MX 2004, CS4에서 업데이트됨

설명

표 폭이 표준 모드로 표시되는지, 아니면 확장 표 모드로 표시되는지 여부를 반환합니다. 표 탭이 레이아웃 모드로 표시되는지 여부에 대한 자세한 내용은 358페이지의 “[dom.getShowLayoutTableTabs\(\)](#)”를 참조하십시오.

인수

없음

반환값

부울 값을 반환합니다. 표 폭이 표준 또는 확장 표 모드로 표시되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getTableExtent()

지원 버전

Dreamweaver 3

설명

이 함수는 선택한 표의 행 및 열 수를 가져옵니다.

인수

없음

반환값

두 개의 정수가 포함된 배열을 반환합니다. 첫 번째 배열 항목은 열의 수이고 두 번째 배열 항목은 행의 수입니다. 표를 선택하지 않은 경우 아무 것도 반환되지 않습니다.

dom.increaseColspan()

지원 버전

Dreamweaver 3

설명

이 함수는 가로 셀을 한 개 합칩니다.

인수

없음

반환값

없음

활성자

435페이지의 “[dom.canIncreaseColspan\(\)](#)”을 참조하십시오.

dom.increaseRowspan()

지원 버전

Dreamweaver 3

설명

이 함수는 세로 셀을 한 개 합칩니다.

인수

없음

반환값

없음

활성자

436페이지의 “[dom.canIncreaseRowspan\(\)](#)”을 참조하십시오.

dom.insertTableColumns()

지원 버전

Dreamweaver 3

설명

이 함수는 현재 표에 지정한 수의 표 열을 삽입합니다.

인수

numberOfCols, bBeforeSelection

- **numberOfCols** 인수는 삽입할 열의 수입니다.
- **bBeforeSelection** 인수는 부울 값입니다. 선택 항목이 포함된 열 앞에 열을 삽입해야 하면 **true**로 설정하고 그렇지 않으면 **false**로 설정합니다.

반환값

없음

활성자

436페이지의 “[dom.canInsertTableColumns\(\)](#)”를 참조하십시오.

dom.insertTableRows()

지원 버전

Dreamweaver 3

설명

이 함수는 현재 표에 지정한 수의 표 행을 삽입합니다.

인수

numberOfRows, bBeforeSelection

- **numberOfRows** 인수는 삽입할 행의 수입니다.
- **bBeforeSelection** 인수는 부울 값입니다. 선택 항목이 포함된 행 위에 행을 삽입해야 하면 **true**로 설정하고 그렇지 않으면 **false**로 설정합니다.

반환값

없음

활성자

436페이지의 “[dom.canInsertTableRows\(\)](#)”를 참조하십시오.

dom.mergeTableCells()

지원 버전

Dreamweaver 3

설명

이 함수는 선택한 표 셀을 병합합니다.

인수

없음

반환값

없음

활성자

437페이지의 “[dom.canMergeTableCells\(\)](#)”를 참조하십시오.

dom.removeAllTableHeights()

지원 버전

Dreamweaver 3

설명

이 함수는 선택한 표에서 HEIGHT 속성을 모두 제거합니다.

인수

없음

반환값

없음

dom.removeAllTableWidths()

지원 버전

Dreamweaver 3

설명

이 함수는 선택한 표에서 WIDTH 속성을 모두 제거합니다.

인수

없음

반환값

없음

dom.removeColumnWidth()

지원 버전

Dreamweaver MX 2004

설명

이 함수는 선택한 한 열에서 WIDTH 속성을 모두 제거합니다.

인수

없음

반환값
없음

dom.selectTable()

지원 버전
Dreamweaver 3

설명
표 전체를 선택합니다.

인수
없음

반환값
없음

활성자
439페이지의 “[dom.canSelectTable\(\)](#)”을 참조하십시오.

dom.setShowTableWidths()

지원 버전
Dreamweaver MX 2004, CS4에서 업데이트됨

설명
표 폭의 표준 또는 확장 표 모드 표시의 설정 및 해제 간을 전환합니다. 이 함수는 다르게 지정되지 않은 경우 현재 문서 및 이후 문서에 대한 값을 설정합니다. 표 탭을 레이아웃 모드로 표시하도록 설정하는 방법에 대한 자세한 내용은 362페이지의 “[dom.setShowLayoutTableTabs\(\)](#)”를 참조하십시오.

인수
bShow

- bShow는 현재 문서가 표준 또는 확장 표 모드일 때 표 폭을 표시할지 여부를 나타내는 부울 인수입니다. bShow가 true이면 폭이 표시됩니다. bShow가 false이면 폭이 표시되지 않습니다.

반환값
없음

dom.setTableCellTag()

지원 버전
Dreamweaver 3

설명
이 함수는 선택한 셀의 태그를 지정합니다.

인수**tdOrTh**

- **tdOrTh** 인수는 "td" 또는 "th"여야 합니다.

반환값

없음

dom.setTableColumns()**지원 버전**

Dreamweaver 3

설명

이 함수는 선택한 표의 열 수를 설정합니다.

인수**numberOfCols**

- **numberOfCols** 인수는 표에 설정할 열의 수를 지정합니다.

반환값

없음

dom.setTableRows()**지원 버전**

Dreamweaver 3

설명

이 함수는 선택한 표의 행 수를 설정합니다.

인수**numberOfCols**

- **numberOfRows** 인수는 선택한 표에 설정할 행의 수를 지정합니다.

반환값

없음

dom.showInsertTableRowsOrColumnsDialog()**지원 버전**

Dreamweaver 3

설명

이 함수는 [행 또는 열 삽입] 대화 상자를 엽니다.

인수

없음

반환값

없음

활성자

436페이지의 “[dom.canInsertTableColumns\(\)](#)” 또는 436페이지의 “[dom.canInsertTableRows\(\)](#)”를 참조하십시오.

dom.splitTableCell()

지원 버전

Dreamweaver 3

설명

이 함수는 현재의 표 셀을 지정한 수의 행 또는 열로 분할합니다. 인수 중 하나를 생략하거나 둘 다 생략하면 [셀 분할] 대화 상자가 나타납니다.

인수**{colsOrRows}, {numberToSplitInto}**

- 선택적인 **colsOrRows** 인수는 "columns" 또는 "rows"여야 합니다.
- 선택적인 **numberToSplitInto** 인수는 셀을 분할할 행 또는 열의 수입니다.

반환값

없음

활성자

440페이지의 “[dom.canSplitTableCell\(\)](#)”을 참조하십시오.

18장: 코드

코드 함수를 사용하면 [코드] 뷰에 나타나는 문서에 대한 작업을 수행할 수 있습니다. 이러한 작업에는 [코드 힌트] 메뉴에 새 메뉴 또는 함수 태그 추가, 문자열 패턴 찾기 및 바꾸기, 문서에서 현재 선택 영역 삭제, 전체 또는 선택한 코드 인쇄, 태그 편집, 선택한 코드에 구문 서식 적용 등이 있습니다.

코드 함수

코드 힌트는 [코드] 뷰에 특정 문자 패턴을 입력하면 Adobe® Dreamweaver® CS4에서 열리는 메뉴입니다. 코드 힌트는 현재 입력하고 있는 문자열을 완성할 수 있는 문자열 목록을 제공하여 입력 시간을 단축할 수 있도록 합니다. 입력하려는 문자열이 메뉴에 나타나면 해당 메뉴로 스크롤하고 Enter 또는 Return 키를 눌러 입력을 마칩니다. 예를 들어 <을 입력하면 팝업 메뉴에 태그 이름 목록이 나타납니다. 나머지 태그 이름을 모두 입력하지 않고 메뉴에서 해당 태그를 선택하면 텍스트에 삽입할 수 있습니다.

CodeHints.xml 파일에서 [코드 힌트] 메뉴를 정의하여 Dreamweaver에 추가할 수 있습니다. CodeHints.xml 파일에 대한 자세한 내용은 **Dreamweaver** 확장을 참조하십시오.

Dreamweaver에 CodeHints.xml 파일의 내용이 로드된 후에는 JavaScript를 통해 새로운 [코드 힌트] 메뉴를 동적으로 추가할 수도 있습니다. 예를 들어 JavaScript 코드는 [바인딩] 패널의 세션 변수 목록을 채웁니다. 동일한 코드를 사용하여 [코드 힌트] 메뉴를 추가할 수 있습니다. 따라서 사용자가 [코드] 뷰에서 **Session**을 입력하면 Dreamweaver에서 세션 변수의 메뉴가 표시됩니다.

CodeHints.xml 파일과 JavaScript API는 코드 힌트 엔진의 유용한 하위 세트를 표시하지만 일부 Dreamweaver 기능에 액세스할 수 없습니다. 예를 들어, Dreamweaver에는 색상 선택기를 열기 위한 JavaScript 후크가 없기 때문에 JavaScript를 사용하여 [속성값] 메뉴를 표현할 수 없습니다. 이 경우 텍스트를 삽입할 수 있는 텍스트 항목 메뉴만 열립니다.

코드 색상 표시를 사용하면 코드 색상 스타일을 지정하여 기존 코드 색상 스키마를 수정하거나 새 코드 색상 스키마를 만들 수 있습니다. Colors.xml 및 코드 색상 스키마 파일을 수정하여 코드 색상 스타일 및 스키마를 지정할 수 있습니다. 이러한 파일에 대한 자세한 내용은 **Dreamweaver** 확장을 참조하십시오.

코드 힌트 및 코드 색상 표시에 대한 The JavaScript API는 다음 함수로 구성됩니다.

dreamweaver.codeHints.addMenu()

지원 버전

Dreamweaver MX

설명

CodeHints.xml 파일에서 새 menu 태그를 동적으로 정의합니다. 패턴과 문서 형식이 같은 menu 태그가 있으면 이 함수는 기존 메뉴에 항목을 추가합니다.

인수

menuGroupId, pattern, labelArray, {valueArray}, {iconArray}, {doctype}, {casesensitive}, {object}, {descriptionArray}, {dismissChars}, {allowWhitespacePrefix}, {restriction}, {type}, {bForcedOnly}, {allowMultipleTimes}, {docURI}

- **menuGroupId** 인수는 menugroup 태그 중 하나의 ID 속성입니다.
- **pattern** 인수는 새 menu 태그의 패턴 속성입니다.
- **labelArray** 인수는 문자열의 배열입니다. 각 문자열은 팝업 메뉴에 있는 단일 메뉴 항목의 텍스트입니다.

- 선택적인 **valueArray** 인수는 문자열 배열이며, **labelArray** 인수와 길이가 같아야 합니다. 사용자가 팝업 메뉴에서 항목을 선택하면 이 배열의 문자열이 사용자의 문서에 삽입됩니다. 삽입할 문자열이 메뉴 레이블과 항상 같으면 이 인수는 **null** 값일 수 있습니다.
- 선택적인 **iconArray** 인수는 문자열 또는 문자열 배열입니다. 한 문자열인 경우에는 Dreamweaver에서 메뉴의 모든 항목에 대해 사용하는 단일 이미지 파일의 URL을 지정합니다. 문자열 배열인 경우에는 **labelArray** 인수와 길이가 같아야 합니다. 각 문자열은 Dreamweaver에서 해당 메뉴 항목의 아이콘으로 사용하는 이미지 파일의 URL입니다. 이 URL은 Dreamweaver Configuration 폴더에 대해 상대적입니다. 이 인수 값이 **null**이면 아이콘 없이 메뉴가 표시됩니다.
- 선택적인 **doctypes** 인수는 이 메뉴가 특정 문서 형식에 대해서만 활성화되도록 지정합니다. **doctypes** 인수는 쉼표로 구분된 문서 형식 ID 목록으로 지정할 수 있습니다. Dreamweaver 문서 형식 목록은 Dreamweaver Configuration/DocumentTypes/MMDocumentTypes.xml 파일을 참조하십시오.
- 선택적인 **casesensitive** 인수는 패턴에서 대/소문자를 구분할지 여부를 지정합니다. **casesensitive** 인수에 사용할 수 있는 값은 **true** 또는 **false**입니다. 이 인수를 생략하는 경우 기본값은 **false**입니다. **casesensitive** 인수 값이 **true**이면 코드 힌트 메뉴가 나타납니다. 이 메뉴는 사용자가 입력한 텍스트가 패턴 속성으로 지정한 패턴과 일치하는 경우에만 표시됩니다. **casesensitive** 인수 값이 **false**이면 패턴이 소문자이고 텍스트가 대문자인 경우에도 메뉴가 표시됩니다.
- **object** 인수는 문자열 이름을 지정합니다. 이 인수는 선택 사항이며 객체 유형이 "정적"인 경우 사용됩니다.
- **descriptionArray** 인수는 코드 힌트에 나타나는 항목에 대해 자세히 설명합니다. 이 인수는 선택 사항입니다.
- **dismissChars** 인수는 사용자가 [코드 힌트] 메뉴를 닫기 위해 입력하는 비표준 문자를 지정합니다. 이 인수는 선택 사항입니다.
- **allowWhitespacePrefix** 인수는 힌트 앞에 공백을 허용하는 부울 값입니다. 이 인수는 선택 사항이며 기본값은 **false**입니다.
- **restriction** 인수는 문자열입니다. 이 인수는 선택 사항이며 지정되지 않은 경우 제한을 적용하지 않습니다. 클라이언트측 및 서버측 언어가 모두 포함된 웹 페이지에서 이 인수를 사용하여 다음 중 하나에 대한 코드 힌트의 사용을 제한할 수 있습니다.
 - 특정 언어 섹션
 - 코드 블록
- **type** 인수는 사용자 인터페이스 메뉴 유형을 정의하는 데 사용됩니다. 이 인수는 선택 사항이며 기본값은 "Enumerated drop down UI"입니다. 가능한 다른 값은 **color**, **font** 및 **url**입니다.
- **bForcedOnly** 인수는 부울 값입니다. **true**이면 키보드 단축키(Ctrl+스페이스바)를 사용하는 경우에만 [코드 힌트] 메뉴가 나타납니다. 이 인수는 선택 사항이며 기본값은 **false**입니다.
- **allowMultipleTimes** 인수는 부울 값입니다. **true**이면 동일한 메뉴가 여러 번 나타날 수 있습니다. 이 인수는 선택 사항이며 기본값은 **false**입니다.
- **docURI** 인수를 사용하면 문서 URI(기본 OS 파일 경로)를 제공하여 특정 문서에 대해 코드 힌트를 제한할 수 있습니다. 이 인수는 선택 사항이며 문서 URI가 지정되지 않은 경우 제한을 적용하지 않습니다.

반환값
없음

예제

사용자가 "myRs"라는 레코드셋을 만들면 다음 코드는 myRS에 대한 메뉴를 만듭니다.

```
dw.codeHints.addMenu(
    "CodeHints_object_methods",           // menu is enabled if object methods are enabled
    "myRS.",                               // pop up menu if user types "myRS."
    new Array("firstName", "lastName"),   // items in pop-up menu for myRS
    new Array("firstName", "lastName"),   // text to actually insert in document
    null,                                  // no icons for this menu
    "ASP_VB, ASP_JS");                   // specific to the ASP doc types
```

dreamweaver.codeHints.addFunction()

지원 버전

Dreamweaver MX

설명

새 **function** 태그를 동적으로 정의합니다. 패턴과 문서 형식이 같은 기존 **function** 태그가 있으면 이 함수는 기존 **function** 태그를 교체합니다.

인수

menuGroupId, pattern, {doctype}, {casesensitive}, {object}, {description}, {icon}, source, {docURI}, {bClassPattern}, {bAddToObjectMethodList}, {restriction}

- menuGroupId 인수는 menugroup 태그의 ID 문자열 속성입니다.
- pattern 인수는 새 **function** 태그의 패턴 속성을 지정하는 문자열입니다.
- 선택적인 doctype 인수는 이 함수가 특정 문서 형식에 대해서만 활성화되도록 지정합니다. doctype 인수는 쉼표로 구분된 문서 형식 ID 목록으로 지정할 수 있습니다. Dreamweaver 문서 형식 목록은 Dreamweaver Configuration/Documenttypes/MMDocumentTypes.xml 파일을 참조하십시오.
- 선택적인 casesensitive 인수는 패턴에서 대/소문자를 구분할지 여부를 지정합니다. **casesensitive** 인수에 사용할 수 있는 값은 **true** 또는 **false**입니다. 이 인수를 생략하는 경우 기본값은 **false**입니다. **casesensitive** 인수 값이 **true**이면 코드 힌트] 메뉴가 나타납니다. 이 메뉴는 사용자가 입력한 텍스트가 패턴 속성으로 지정한 패턴과 일치하는 경우에만 표시됩니다. **casesensitive** 값이 **false**이면 패턴이 소문자이고 텍스트가 대문자인 경우에도 메뉴가 표시됩니다.
- object 인수는 문자열 이름을 지정합니다. 이 인수는 선택 사항이며 객체 유형이 "정적"인 경우 사용됩니다.
- description 인수에는 함수에 대한 자세한 설명이 포함됩니다. 이 인수는 선택 사항입니다.
- icon 인수는 함수 드롭다운에서 사용할 사용자 정의 아이콘 경로를 지정합니다. 이 인수는 선택 사항입니다.
- source 인수에는 코드 힌트의 두 번째 열에 표시되는 값이 포함됩니다. 이 인수의 기본값은 **empty**입니다.
- docURI 인수를 사용하면 문서 URI(기본 OS 파일 경로)를 제공하여 특정 문서에 대해 코드 힌트를 제한할 수 있습니다. 이 인수는 선택 사항이며 문서 URI가 지정되지 않은 경우 제한을 적용하지 않습니다.
- bClassPattern 인수는 **true**로 설정되면 함수가 "클래스" 인스턴스에 속하고 정적이 아님을 나타냅니다. 기본값은 **false**입니다. 이 인수는 선택 사항입니다.
- bAddToObjectMethodList 인수는 **true**로 설정되면 사용자가 정적 함수의 목록을 추가할 수 있습니다. 기본값은 **true**입니다. 이 인수는 선택 사항입니다.
- restriction 인수는 문자열입니다. 이 인수는 선택 사항이며 지정되지 않은 경우 제한을 적용하지 않습니다. 클라이언트측 및 서버측 언어가 모두 포함된 웹 페이지에서 이 인수를 사용하여 다음 중 하나에 대한 코드 힌트의 사용을 제한할 수 있습니다.
 - 특정 언어 섹션
 - 코드 블록

반환값

없음

예제

dw.codeHints.addFunction() 함수의 다음 예제에서는 함수 이름 패턴 **out.newLine()**을 [코드 힌트] 메뉴 그룹 **CodeHints_Object_Methods**에 추가하고 JSP 문서 형식에 대해서만 활성화되도록 합니다.

```
dw.codeHints.addFunction(
    "CodeHints_Object_Methods",
    "out.newLine()",
    "JSP")
```

dreamweaver.codeHints.resetMenu()

지원 버전

Dreamweaver MX

설명

지정된 **menu** 태그 또는 **function** 태그의 상태를 Dreamweaver에서 **CodeHints.xml** 파일을 읽은 직후의 상태로 다시 설정합니다. 즉, 이 함수를 호출하면 **addMenu()** 및 **addFunction()** 함수에 대한 이전 호출의 결과가 지워집니다.

인수

menuGroupId, **pattern**, {**doctype**s}

- **menuGroupId** 인수는 **menugroup** 태그의 ID 문자열 속성입니다.
- **pattern** 인수는 다시 설정할 새 **menu** 또는 **function** 태그의 패턴 속성을 지정하는 문자열입니다.
- 선택적인 **doctype**s 인수는 이 메뉴가 특정 문서 형식에 대해서만 활성화되도록 지정합니다. **doctype**s 인수는 쉼표로 구분된 문서 형식 ID 목록으로 지정할 수 있습니다. Dreamweaver 문서 형식 목록은 Dreamweaver Configuration/Documenttypes/MMDocumentTypes.xml 파일을 참조하십시오.

반환값

없음

예제

JavaScript 코드를 사용하면 사용자 정의 세션 변수가 포함된 [코드 힌트] 메뉴를 구성할 수 있습니다. 세션 변수의 목록이 변경될 때마다 해당 코드는 메뉴를 업데이트해야 합니다. 코드에서 세션 변수의 새 목록을 메뉴로 로드하려면 이전 목록을 제거해야 합니다. 이 함수를 호출하면 이전 세션 변수가 제거됩니다.

dreamweaver.codeHints.showCodeHints()

지원 버전

Dreamweaver MX

설명

사용자가 [편집] > [코드 힌트 표시] 메뉴 항목을 열면 Dreamweaver에서 이 함수를 호출합니다. 이 함수는 [코드] 뷰의 현재 선택 영역 위치에서 [코드 힌트] 메뉴를 엽니다.

인수

없음

반환값

없음

예제

다음 예제에서는 [코드] 뷰에 있을 때 문서의 현재 삽입점에 [코드 힌트] 메뉴를 엽니다.

```
dw.codeHints.showCodeHints()
```

dreamweaver.reloadCodeColoring()

설명

Dreamweaver Configuration/Code Coloring 폴더에서 코드 색상 표시 파일을 다시 로드합니다.

인수

없음

반환값

없음

예제

```
dreamweaver.reloadCodeColoring()
```

찾기 및 바꾸기 함수

찾기 및 바꾸기 함수는 찾기 및 바꾸기 작업을 처리합니다. 이러한 작업에는 사용자 상호 작용이 필요 없는 복잡한 바꾸기 작업 및 검색 패턴의 다음 인스턴스 찾기와 같은 기본 기능이 포함됩니다.

dreamweaver.findNext()

지원 버전

Dreamweaver 3, Dreamweaver MX 2004에서 수정됨

설명

389페이지의 “[dreamweaver.setUpFind\(\)](#)”, 387페이지의 “[dreamweaver.setUpComplexFind\(\)](#)” 또는 [찾기] 대화 상자에 서 사용자가 이미 지정한 검색 문자열의 다음 인스턴스를 찾고 문서에서 인스턴스를 선택합니다.

인수

{bUseLastSetupSearch}

- 선택적인 **bUseLastSetupSearch** 인수는 부울 값입니다. **bUseLastSetupSearch** 값이 true(주어진 인수가 없는 경우 기본 값)이면 이 함수는 이전에 [dreamweaver.setupComplexFind\(\)](#) 함수 또는 [dreamweaver.setupComplexFindReplace\(\)](#) 함수를 호출 하여 지정한 매개 변수를 사용하여 다음 찾기 작업을 수행합니다. **bUseLastSetupSearch** 값을 false로 설정하면 이 함수는 이전에 설정된 검색을 무시하고 문서에서 현재 선택된 텍스트의 다음 인스턴스를 검색합니다.

반환값

없음

활성자

444페이지의 “[dreamweaver.canFindNext\(\)](#)”를 참조하십시오.

dreamweaver.replace()

지원 버전

Dreamweaver 3

설명

현재 선택 영역이 389페이지의 “[dreamweaver.setUpFindReplace\(\)](#)”, 388페이지의 “[dreamweaver.setUpComplexFindReplace\(\)](#)” 또는 [바꾸기] 대화 상자에서 사용자가 지정한 검색 조건과 일치하는지 확인한 다음 선택 영역을 검색 요청에 지정된 대체 텍스트로 바꿉니다.

인수

없음

반환값

없음

dreamweaver.replaceAll()

지원 버전

Dreamweaver 3

설명

389페이지의 “[dreamweaver.setUpFindReplace\(\)](#)”, 388페이지의 “[dreamweaver.setUpComplexFindReplace\(\)](#)” 또는 [바꾸기] 대화 상자에서 사용자가 지정한 검색 조건과 일치하는 현재 문서의 각 단원을 지정된 대체 내용으로 바꿉니다.

인수

없음

반환값

없음

dreamweaver.setUpComplexFind()

지원 버전

Dreamweaver 3

설명

지정된 XML 쿼리를 로드하여 텍스트 또는 태그에 대한 고급 검색을 준비합니다.

인수

xmlQueryString

- **xmlQueryString** 인수는 dwquery로 시작하고 /dwquery로 끝나는 XML 코드의 문자열입니다. 적절한 형식의 문자열을 가져오려면 [찾기] 대화 상자에서 쿼리를 설정하고 [쿼리 저장] 버튼을 클릭한 다음 텍스트 편집기에서 쿼리 파일을 열어 dwquery 태그의 시작 부분부터 /dwquery 태그의 끝 부분까지 모두 복사합니다.

참고: 쿼리에서 백슬래시 문자(\) 등의 일부 특수 문자는 이스케이프 처리해야 하므로 쿼리에서 백슬래시를 사용하려면 \\를 사용해야 합니다.

반환값

없음

예제

다음 예제의 첫 줄에서는 태그 검색을 설정하고 검색 범위를 현재 문서로 지정합니다. 둘째 줄에서는 검색 작업을 수행합니다.

```
dreamweaver.setUpComplexFind('<dwquery><queryparams matchcase="false" ~
ignorewhitespace="true" useregexp="false"/><find>~
<tag qname="a"><attribute qname="href" qcompare="=" qvalue="#">~
  </attribute><attribute qname="onMouseOut" qcompare="=" qvalue="" qnegate="true">~
  </attribute></tag></find></dwquery>');
dw.findNext();
```

dreamweaver.setUpComplexFindReplace()

지원 버전

Dreamweaver 3

설명

지정된 XML 쿼리를 로드하여 텍스트 또는 태그에 대한 고급 검색을 준비합니다.

인수

xmlQueryString

- **xmlQueryString** 인수는 dwquery 태그로 시작하고 /dwquery 태그로 끝나는 XML 코드의 문자열입니다. 적절한 형식의 문자열을 가져오려면 [찾기] 대화 상자에서 쿼리를 설정하고 [쿼리 저장] 버튼을 클릭한 다음 텍스트 편집기에서 쿼리 파일을 열어 dwquery 태그의 시작 부분부터 /dwquery 태그의 끝 부분까지 모두 복사합니다.

참고: 쿼리에서 백슬래시 문자(\) 등의 일부 특수 문자는 이스케이프 처리해야 하므로 쿼리에서 백슬래시를 사용하려면 \\를 사용해야 합니다.

반환값

없음

예제

다음 예제의 첫 번째 명령문에서는 태그 검색을 설정하고 검색 범위를 파일 네 개로 지정합니다. 두 번째 명령문에서는 검색 및 바꾸기 작업을 수행합니다.

```
dreamweaver.setUpComplexFindReplace('<dwquery><queryparams ~
matchcase="false" ignorewhitespace="true" useregexp="false"/>~
<find><tag qname="a"><attribute qname="href" qcompare="=" qvalue="#">~
</attribute><attribute qname="onMouseOut" ~qcompare="=" qvalue="" qnegate="true">~
</attribute></tag></find><replace action="setAttribute" param1="onMouseOut" ~
param2="this.style.color='#000000';this.style.~
fontWeight='normal'"/></dwquery>');
dw.replaceAll();
```

dreamweaver.setUpFind()

지원 버전

Dreamweaver 3

설명

그 다음 `dreamweaver.findNext()` 작업의 검색 매개 변수를 정의하여 텍스트나 HTML 소스 검색을 준비합니다.

인수

searchObject

searchObject 인수는 다음 속성을 정의할 수 있는 객체입니다.

- **searchString**은 검색할 텍스트입니다.
- **searchSource** 속성은 HTML 소스를 검색할지 여부를 나타내는 부울 값입니다.
- 선택적인 **{matchCase}** 속성은 검색에서 대/소문자를 구분할지 여부를 나타내는 부울 값입니다. 이 속성이 명시적으로 설정되지 않은 경우 기본값은 `false`입니다.
- 선택적인 **{ignoreWhitespace}** 속성은 공백을 무시할지 여부를 나타내는 부울 값입니다. **ignoreWhitespace** 속성의 기본값은 **useRegularExpressions** 속성 값이 `true`이면 `false`가 되고, **useRegularExpressions** 속성이 `false`이면 `true`가 됩니다.
- **{useRegularExpressions}** 속성은 **searchString** 속성에 정규식이 사용되는지 여부를 나타내는 부울 값입니다. 이 속성이 명시적으로 설정되지 않은 경우 기본값은 `false`입니다.

반환값

없음

예제

다음 코드는 **searchObject** 객체를 만드는 세 가지 방법을 보여 줍니다.

```
var searchParams;
searchParams.searchString = 'bgcolor="#FFCCFF"';
searchParams.searchSource = true;
dreamweaver.setUpFind(searchParams);
var searchParams = {searchString: 'bgcolor="#FFCCFF"', searchSource: true};
dreamweaver.setUpFind(searchParams);
dreamweaver.setUpFind({searchString: 'bgcolor="#FFCCFF"', searchSource: true});
```

dreamweaver.setUpFindReplace()

지원 버전

Dreamweaver 3

설명

그 다음 `dreamweaver.replace()` 또는 `dreamweaver.replaceAll()` 작업의 범위와 검색 매개 변수를 정의하여 텍스트나 HTML 소스 검색을 준비합니다.

인수

searchObject

searchObject 인수는 다음 속성을 정의할 수 있는 객체입니다.

- **searchString** 속성은 검색할 텍스트입니다.
- **replaceString** 속성은 선택 영역을 바꿀 텍스트입니다.
- **searchSource** 속성은 HTML 소스를 검색할지 여부를 나타내는 부울 값입니다.
- 선택적인 **{matchCase}** 속성은 검색에서 대/소문자를 구분할지 여부를 나타내는 부울 값입니다. 이 속성이 명시적으로 설정되지 않은 경우 기본값은 **false**입니다.
- 선택적인 **{ignoreWhitespace}** 속성은 공백을 무시할지 여부를 나타내는 부울 값입니다. **ignoreWhitespace** 속성의 기본값은 **useRegularExpressions** 속성 값이 **true**이면 **false**가 되고, **useRegularExpressions** 속성 값이 **false**이면 **true**가 됩니다.
- **{useRegularExpressions}** 속성은 **searchString** 속성에 정규식이 사용되는지 여부를 나타내는 부울 값입니다. 이 속성이 명시적으로 설정되지 않은 경우 기본값은 **false**입니다.

반환값

없음

예제

다음 코드는 **searchObject** 객체를 만드는 세 가지 방법을 보여 줍니다.

```
var searchParams;
searchParams.searchString = 'bgcolor="#FFCCFF"';
searchParams.replaceString = 'bgcolor="#CCFFCC"';
searchParams.searchSource = true;
dreamweaver.setUpFindReplace(searchParams);

var searchParams = {searchString: 'bgcolor="#FFCCFF"', replaceString: 'bgcolor="#CCFFCC"',
    searchSource: true};
dreamweaver.setUpFindReplace(searchParams);

dreamweaver.setUpFindReplace({searchString: 'bgcolor="#FFCCFF"',
    replaceString: 'bgcolor="#CCFFCC"', searchSource: true});
```

dreamweaver.showFindDialog()

지원 버전

Dreamweaver 3

설명

[찾기] 대화 상자를 엽니다.

인수

없음

반환값

없음

활성자

448페이지의 “[dreamweaver.canShowFindDialog\(\)](#)”를 참조하십시오.

dreamweaver.showFindReplaceDialog()

지원 버전

Dreamweaver 3

설명

[바꾸기] 대화 상자를 엽니다.

인수

없음

반환값

없음

활성자

448페이지의 “[dreamweaver.canShowFindDialog\(\)](#)”를 참조하십시오.

일반 편집 함수

일반 편집 함수는 [문서] 윈도우에서 처리합니다. 이러한 함수는 텍스트, HTML 및 객체를 삽입하고, 글꼴 및 문자 마크업을 적용, 변경 및 제거하고, 태그 및 속성을 수정하는 등의 작업을 수행합니다.

dom.applyCharacterMarkup()

지원 버전

Dreamweaver 3

설명

지정된 유형의 문자 마크업을 선택 영역에 적용합니다. 선택 영역이 삽입점인 경우 지정된 문자 마크업은 그 다음에 입력된 모든 텍스트에 적용됩니다.

인수

tagName

- **tagName** 인수는 문자 마크업과 연관된 태그 이름입니다. 이 인수는 "b", "cite", "code", "dfn", "em", "i", "kbd", "samp", "s", "strong", "tt", "u" 또는 "var" 문자열 중 하나여야 합니다.

반환값

없음

dom.applyFontMarkup()

지원 버전

Dreamweaver 3

설명

FONT 태그 및 지정된 속성과 해당 값을 현재 선택 영역에 적용합니다.

인수

attribute, value

- **attribute** 인수는 "face", "size" 또는 "color"여야 합니다.
- **value** 인수는 속성에 할당될 값입니다(예: "Arial, Helvetica, sans-serif", "5" 또는 "#FF0000").

반환값

없음

dom.deleteSelection()

지원 버전

Dreamweaver 3

설명

문서에서 선택 영역을 삭제합니다.

인수

없음

반환값

없음

dom.editAttribute()

지원 버전

Dreamweaver 3

설명

지정된 문서 속성을 편집하는 데 적합한 인터페이스를 표시합니다. 대부분의 경우 이 인터페이스는 대화 상자입니다. 이 함수는 활성 문서에만 사용할 수 있습니다.

인수

attribute

- **attribute**는 편집할 태그 속성을 지정하는 문자열입니다.

반환값

없음

dom.exitBlock()

지원 버전

Dreamweaver 3

설명

삽입점을 모든 블록 요소의 외부에 남겨두고 현재 단락이나 머리글 블록을 끝냅니다.

인수

없음

반환값

없음

dom.getCharSet()

지원 버전

Dreamweaver 4

설명

문서에 있는 메타 태그의 charset 속성을 반환합니다.

인수

없음

반환값

문서의 인코딩 ID를 반환합니다. 예를 들어, 영어(Latin1) 문서의 경우 이 함수는 iso-8859-1을 반환합니다.

dom.getFontMarkup()

지원 버전

Dreamweaver 3

설명

현재 선택 영역의 FONT 태그에 지정된 속성의 값을 가져옵니다.

인수

attribute

- attribute 인수는 "face", "size" 또는 "color"여야 합니다.

반환값

지정된 속성의 값이 포함된 문자열을 반환하거나, 속성이 설정되지 않은 경우에는 빈 문자열을 반환합니다.

dom.getLineFromOffset()

지원 버전

Dreamweaver MX

설명

파일의 텍스트(HTML 또는 JavaScript 코드)에서 특정 문자 오프셋의 행 번호를 찾습니다.

인수

offset

- **offset** 인수는 파일의 시작 부분에서부터 문자 위치를 나타내는 정수입니다.

반환값

문서의 행 번호를 나타내는 정수를 반환합니다.

dom.getLinkHref()

지원 버전

Dreamweaver 3

설명

현재 선택 영역 주위에 있는 링크를 가져옵니다. 이 함수는 링크를 찾을 때까지 현재 노드의 부모 노드 및 부모 노드의 상위 노드를 반복하고 링크에서 `getAttribute("HREF")`를 호출하는 것과 같습니다.

인수

없음

반환값

링크된 파일의 이름을 포함하는 문자열을 반환합니다. 이 이름은 `file:// URL`로 표시됩니다.

dom.getLinkTarget()

지원 버전

Dreamweaver 3

설명

현재 선택 영역 주위에 있는 링크의 대상을 가져옵니다. 이 함수는 링크를 찾을 때까지 현재 노드의 부모 노드 및 부모 노드의 상위 노드를 반복하고 링크에서 `getAttribute("TARGET")` 함수를 호출하는 것과 같습니다.

인수

없음

반환값

링크에 대한 TARGET 속성의 값이 포함된 문자열을 반환하거나, 대상이 지정되지 않은 경우에는 빈 문자열을 반환합니다.

dom.getListTag()

지원 버전

Dreamweaver 3

설명

선택한 목록의 스타일을 가져옵니다.

인수

없음

반환값

목록("ul", "ol" 또는 "dl")과 연관된 태그를 포함하는 문자열을 반환하거나, 태그가 목록과 연관되지 않은 경우에는 빈 문자열을 반환합니다. 이 값은 항상 소문자로 반환됩니다.

dom.getTextAlignment()

지원 버전

Dreamweaver 3

설명

선택 영역을 포함하는 블록의 정렬을 가져옵니다.

인수

없음

반환값

블록과 연관된 태그에 대한 ALIGN 속성의 값이 포함된 문자열을 반환하거나, 태그에 ALIGN 속성이 설정되지 않은 경우에는 빈 문자열을 반환합니다. 이 값은 항상 소문자로 반환됩니다.

dom.getTextFormat()

지원 버전

Dreamweaver 3

설명

선택한 텍스트의 블록 형식을 가져옵니다.

인수

없음

반환값

텍스트와 연관된 블록 태그가 포함된 문자열(예: "p", "h1", "pre" 등)을 반환하거나, 선택 영역과 연관된 블록 태그가 없는 경우에는 빈 문자열을 반환합니다. 이 값은 항상 소문자로 반환됩니다.

dom.hasCharacterMarkup()

지원 버전

Dreamweaver 3

설명

선택 영역에 지정된 문자 마크업이 이미 있는지 여부를 확인합니다.

인수

markupTagName

- **markupTagName** 인수는 확인하려는 태그의 이름입니다. 이 인수는 "b", "cite", "code", "dfn", "em", "i", "kbd", "samp", "s", "strong", "tt", "u" 또는 "var" 문자열 중 하나여야 합니다.

반환값

전체 선택 영역에 지정된 문자 마크업이 있는지 여부를 나타내는 부울 값을 반환합니다. 선택 영역의 일부에만 지정된 마크업이 있는 경우에는 false 값이 반환됩니다.

dom.indent()

지원 버전

Dreamweaver 3

설명

BLOCKQUOTE 태그를 사용하여 선택 영역을 들여쓰습니다. 선택 영역이 목록 항목인 경우 이 함수는 선택한 항목을 중첩된 목록으로 변환하여 선택 영역을 들여쓰습니다. 이 중첩된 목록은 외부 목록과 유형이 같고 한 항목(원본 선택 영역)을 포함합니다.

인수

없음

반환값

없음

dom.insertHTML()

지원 버전

Dreamweaver 3

설명

문서의 현재 삽입점에 HTML 내용을 삽입합니다.

인수

contentToInsert, {bReplaceCurrentSelection}

- **contentToInsert** 인수는 삽입할 내용입니다.

- 선택적인 **bReplaceCurrentSelection** 인수는 현재 선택 영역을 해당 내용으로 바꿀지 여부를 나타내는 부울 값입니다. **bReplaceCurrentSelection** 인수 값이 **true**이면 현재 선택 영역이 해당 내용으로 바뀌고, **false**이면 현재 선택 영역 뒤에 해당 내용이 삽입됩니다.

반환값

없음

예제

다음 코드에서는 현재 문서에 HTML 문자열 `<b>130</b>`를 삽입합니다.

```
var theDOM = dw.getDocumentDOM();
theDOM.insertHTML('<b>130</b>');
```

결과는 [문서] 윈도우에 나타납니다.

dom.insertObject()

지원 버전

Dreamweaver 3

설명

지정된 객체를 삽입하며 필요한 경우 사용자에게 매개 변수를 입력하도록 요청합니다.

인수**objectName**

- **objectName** 인수는 Configuration/Objects 폴더에 있는 객체의 이름입니다.

반환값

없음

예제

`dom.insertObject('Button')` 함수를 호출하면 활성 문서에서 현재 선택 영역 뒤에 양식 버튼이 삽입됩니다. 아무 것도 선택하지 않은 경우에는 현재 삽입점에 버튼이 삽입됩니다.

참고: 객체 파일을 다른 폴더에 저장할 수 있지만 파일 이름은 고유하게 사용해야 합니다. **Button.htm**이라는 파일이 **Forms** 폴더와 **MyObjects** 폴더에 모두 있으면 **Dreamweaver**에서는 이 두 파일을 구별하지 못합니다.

dom.insertText()

지원 버전

Dreamweaver 3

설명

문서의 현재 삽입점에 텍스트 내용을 삽입합니다.

인수**contentToInsert, {bReplaceCurrentSelection}**

- **contentToInsert** 인수는 삽입할 내용입니다.

- 선택적인 **bReplaceCurrentSelection** 인수는 현재 선택 영역을 해당 내용으로 바꿀지 여부를 나타내는 부울 값입니다. **bReplaceCurrentSelection** 인수 값이 **true**이면 현재 선택 영역이 해당 내용으로 바뀌고, **false**이면 현재 선택 영역 뒤에 해당 내용이 삽입됩니다.

반환값

없음

예제

다음 코드에서는 130 텍스트를 현재 문서에 삽입합니다.

```
var theDOM = dreamweaver.getDocumentDOM();
theDOM.insertText('<b>130</b>');
```

결과는 [문서] 윈도우에 나타납니다.

dom.newBlock()

지원 버전

Dreamweaver 3

설명

현재 선택 영역을 포함하는 블록과 태그 및 속성이 같은 새 블록을 만들거나 포인터가 모든 블록 외부에 있는 경우 새 단락을 만듭니다.

인수

없음

반환값

없음

예제

현재 선택 영역이 가운데 정렬된 단락 내에 있는 경우 **dom.newBlock()** 함수를 호출하면 현재 단락 뒤에 <p align="center">가 삽입됩니다.

dom.notifyFlashObjectChanged()

지원 버전

Dreamweaver 4

설명

현재 Flash 객체 파일이 변경되었음을 Dreamweaver에 알려 줍니다. Dreamweaver에서는 미리 보기 화면을 업데이트하고 원본 크기의 폭과 높이 비율을 유지하면서 필요에 따라 크기를 조절합니다.

인수

없음

반환값

없음

dom.outdent()

지원 버전

Dreamweaver 3

설명

선택 영역을 내어줍니다.

인수

없음

반환값

없음

dom.removeCharacterMarkup()

지원 버전

Dreamweaver 3

설명

선택 영역에서 지정된 유형의 문자 마크업을 제거합니다.

인수

tagName

- **tagName** 인수는 문자 마크업과 연관된 태그 이름입니다. 이 인수는 "b", "cite", "code", "dfn", "em", "i", "kbd", "samp", "s", "strong", "tt", "u" 또는 "var" 문자열 중 하나여야 합니다.

반환값

없음

dom.removeFontMarkup()

지원 버전

Dreamweaver 3

설명

FONT 태그에서 지정된 속성과 해당 값을 제거합니다. 속성을 제거할 때 FONT 태그만 남겨 놓으면 FONT 태그도 제거됩니다.

인수

attribute

- **attribute** 인수는 "face", "size" 또는 "color"여야 합니다.

반환값

없음

dom.resizeSelection()

지원 버전

Dreamweaver 3

설명

지정된 치수로 선택한 객체의 크기를 조절합니다.

인수

newWidth, newHeight

- **newWidth** 인수는 함수에서 선택된 객체를 설정할 새 폭을 지정합니다.
- **newHeight** 인수는 함수에서 선택된 객체를 설정할 새 높이를 지정합니다.

반환값

없음

dom.setAttributeWithErrorChecking()

지원 버전

Dreamweaver 3

설명

현재 선택 영역에 지정된 값으로 지정된 속성을 설정하고, 값의 유형이 잘못되었거나 범위를 벗어나는 경우 사용자에게 알려 주는 메시지를 표시합니다. 이 함수는 활성 문서에만 사용할 수 있습니다.

인수

attribute, value

- **attribute** 인수는 현재 선택 영역에 설정할 속성을 지정합니다.
- **value** 인수는 속성에 설정할 값을 지정합니다.

반환값

없음

dom.setLinkHref()

지원 버전

Dreamweaver 3

설명

선택 영역을 하이퍼텍스트 링크로 만들거나 현재 선택 영역 주위에 지정된 HREF 태그의 URL 값을 변경합니다.

인수**linkHref**

- **linkHref** 인수는 링크를 구성하는 URL(문서의 상대 경로, 루트의 상대 경로 또는 절대 URL)입니다. 이 인수를 생략하면 [HTML 파일 선택] 대화 상자가 나타납니다.

반환값

없음

활성자

439페이지의 “[dom.canSetLinkHref\(\)](#)”를 참조하십시오.

dom.setLinkTarget()

지원 버전

Dreamweaver 3

설명

현재 선택 영역 주위에 지정된 링크의 대상을 설정합니다. 이 함수는 링크를 찾을 때까지 현재 노드의 부모 노드 및 부모 노드의 상위 노드를 반복하고 링크에서 `setAttribute("TARGET")` 함수를 호출하는 것과 같습니다.

인수**{linkTarget}**

- 선택적인 **linkTarget** 인수는 프레임 이름, 윈도우 이름 또는 예약된 대상("_self", "_parent", "_top" 또는 "_blank")중 하나를 나타내는 문자열입니다. 이 인수를 생략하면 [대상 설정] 대화 상자가 나타납니다.

반환값

없음

dom.setListBoxKind()

지원 버전

Dreamweaver 3

설명

선택한 SELECT 메뉴의 종류를 변경합니다.

인수**kind**

- **kind** 인수는 "menu" 또는 "list box"여야 합니다.

반환값

없음

dom.showListPropertiesDialog()

지원 버전

Dreamweaver 3

설명

[목록 속성] 대화 상자를 엽니다.

인수

없음

반환값

없음

활성자

439페이지의 “[dom.canShowListPropertiesDialog\(\)](#)”를 참조하십시오.

dom.setListTag()

지원 버전

Dreamweaver 3

설명

선택한 목록의 스타일을 설정합니다.

인수

listTag

- listTag 인수는 목록과 연관된 태그입니다. 이 인수는 "ol", "ul", "dl" 또는 빈 문자열이어야 합니다.

반환값

없음

dom.setTextAlignment()

지원 버전

Dreamweaver 3

설명

선택 영역이 포함된 블록의 ALIGN 속성을 지정된 값으로 설정합니다.

인수

alignValue

- alignValue 인수는 "left", "center" 또는 "right"여야 합니다.

반환값
없음

dom.setTextFieldKind()

지원 버전
Dreamweaver 3

설명
선택한 텍스트 필드의 형식을 설정합니다.

인수
fieldType

- **fieldType** 인수는 "input", "textarea" 또는 "password"여야 합니다.

반환값
없음

dom.setTextFormat()

지원 버전
Dreamweaver 4

설명
선택한 텍스트의 블록 형식을 설정합니다.

인수
blockFormat

- **blockFormat** 인수는 ""(형식이 없는 경우), "p", "h1", "h2", "h3", "h4", "h5", "h6" 또는 "pre" 형식 중 하나를 지정하는 문자열입니다.

반환값
없음

dom.showFontColorDialog()

지원 버전
Dreamweaver 3

설명
[색상 선택기] 대화 상자를 엽니다.

인수
없음

반환값

없음

dreamweaver.deleteSelection()

지원 버전

Dreamweaver 3

설명

활성 문서나 [사이트] 패널에서 선택 영역을 삭제합니다. Macintosh에서는 대화 상자나 부동 패널에서 포커스가 있는 텍스트 상자를 삭제합니다.

인수

없음

반환값

없음

활성자

443페이지의 “[dreamweaver.canDeleteSelection\(\)](#)”을 참조하십시오.

dreamweaver.editFontList()

지원 버전

Dreamweaver 3

설명

[글꼴 목록 편집] 대화 상자를 엽니다.

인수

없음

반환값

없음

dreamweaver.getFontList()

지원 버전

Dreamweaver 3

설명

텍스트 속성 관리자와 [스타일 정의] 대화 상자에 나타나는 모든 글꼴 그룹의 목록을 가져옵니다.

인수

없음

반환값

글꼴 목록의 각 항목을 나타내는 문자열 배열을 반환합니다.

예제

Dreamweaver의 기본 설치에 대해 `dreamweaver.getFontList()` 함수를 호출하면 다음 항목을 포함하는 배열이 반환됩니다.

- "Arial, Helvetica, sans-serif"
- "Times New Roman, Times, serif"
- "Courier New, Courier, mono"
- "Georgia, Times New Roman, Times, serif"
- "Verdana, Arial, Helvetica, sans-serif"

dreamweaver.getFontStyles()

지원 버전

Dreamweaver 4

설명

지정된 트루타입 글꼴이 지원하는 스타일을 반환합니다.

인수

fontName

- **fontName** 인수는 글꼴 이름을 포함하는 문자열입니다.

반환값

글꼴이 지원하는 내용을 나타내는 세 가지 부울 값의 배열을 반환합니다. 첫 번째 값은 글꼴이 굵은 글꼴을 지원하는지 여부를 나타내고, 두 번째 값은 기울임체 글꼴을 지원하는지 여부, 세 번째 값은 굵은 글꼴과 기울임체 글꼴을 모두 지원하는지 여부를 나타냅니다.

dreamweaver.getKeyState()

지원 버전

Dreamweaver 3

설명

지정된 수정자 키를 눌렀는지 여부를 확인합니다.

인수

key

- **key** 인수는 "Cmd", "Ctrl", "Alt" 또는 "Shift" 값 중 하나여야 합니다. Windows에서 "Cmd"와 "Ctrl"은 Control 키를 나타내고, Macintosh에서 "Alt"는 Option 키를 나타냅니다.

반환값

키를 눌렀는지 여부를 나타내는 부울 값을 반환합니다.

예제

다음 코드에서는 작업을 수행하기 전에 Shift 키와 Ctrl 키(Windows) 또는 Shift 키와 Command 키(Macintosh)를 모두 눌렀는지 확인합니다.

```
if (dw.getKeyState("Shift") && dw.getKeyState("Cmd")){
    // execute code
}
```

dreamweaver.getNaturalSize()

지원 버전

Dreamweaver 4

설명

그래픽 객체의 폭과 높이를 반환합니다.

인수

url

- **url** 인수는 치수를 반환할 그래픽 객체를 가리킵니다. Dreamweaver에서는 이 객체(GIF, JPEG, PNG, Flash 및 Shockwave)를 지원해야 합니다. getNaturalSize() 함수에 대한 인수로 제공되는 URL은 로컬 파일을 가리키는 절대 URL이어야 하며, 상대 URL이 될 수 없습니다.

반환값

객체의 폭을 정의하는 첫 번째 정수와 높이를 정의하는 두 번째 정수로 구성된 배열을 반환합니다.

dreamweaver.getSystemFontList()

지원 버전

Dreamweaver 4

설명

시스템의 글꼴 목록을 반환합니다. 이 함수는 모든 글꼴이나 트루타입 글꼴만 가져올 수 있습니다.

인수

fontTypes

- fontTypes 인수는 all 또는 TrueType 값을 포함하는 문자열입니다.

반환값

전체 글꼴 이름을 포함하는 문자열의 배열을 반환하거나, 글꼴이 없는 경우에는 null을 반환합니다.

인쇄 함수

인쇄 함수를 사용하면 [코드] 뷰에서 코드를 인쇄할 수 있습니다.

dreamweaver.printCode()

지원 버전

Dreamweaver MX

설명

Windows에서는 [코드] 뷰에서 코드의 전체 또는 선택한 부분을 인쇄하고, Macintosh에서는 코드의 전체 또는 일정 범위의 페이지를 인쇄합니다.

인수

showPrintDialog, document

- **showPrintDialog** 인수는 true 또는 false입니다. 이 인수가 true로 설정된 경우 Windows에서는 dreamweaver.PrintCode() 함수가 [인쇄] 대화 상자를 표시하여 모든 텍스트나 선택한 텍스트를 인쇄할지 여부를 묻습니다. Macintosh에서는 dreamweaver.PrintCode() 함수가 [인쇄] 대화 상자를 표시하여 모든 텍스트나 일정 범위의 페이지를 인쇄할지 여부를 묻습니다.

이 인수가 false로 설정된 경우 dreamweaver.PrintCode()는 사용자의 이전 선택 영역을 사용합니다. 기본값은 true입니다.

- **document** 인수는 인쇄할 문서의 DOM입니다. 문서의 DOM을 얻는 방법에 대한 자세한 내용은 237페이지의 “[dreamweaver.getDocumentDOM\(\)](#)”을 참조하십시오.

반환값

불을 값을 반환합니다. 코드를 인쇄할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

다음 예제에서는 dw.PrintCode()를 호출하여 사용자 문서에 대한 [인쇄] 대화 상자를 시작합니다. 함수가 false를 반환하면 코드에서는 사용자에게 인쇄 요청을 실행할 수 없음을 알려 주는 경고 메시지를 표시합니다.

```
var theDOM = dreamweaver.getDocumentDOM("document");
if(!dreamweaver.PrintCode(true, theDOM))
{
    alert("Unable to execute your print request!");
}
```

릭 태그 편집기 함수

릭 태그 편집기 함수는 현재 선택 영역의 내부 및 주변의 태그에 대해 작업을 수행하며, 계층 구조의 태그를 모두 제거하고, 새 태그로 선택 영역을 둘러싸고, 릭 태그 편집기를 표시하여 사용자가 태그의 특정 속성을 편집할 수 있도록 합니다.

dom.selectChild()

지원 버전

Dreamweaver 3

설명

현재 선택 영역의 자식을 선택합니다. 이 함수를 호출하면 [문서] 윈도우의 아래쪽에 있는 태그 선택기에서 오른쪽의 다음 태그를 선택할 때와 동일한 결과가 나타납니다.

인수
없음

반환값
없음

dom.selectParent()

지원 버전
Dreamweaver 3

설명
현재 선택 영역의 부모를 선택합니다. 이 함수를 호출하면 [문서] 윈도우의 아래쪽에 있는 태그 선택기에서 왼쪽의 다음 태그를 선택할 때와 동일한 결과가 나타납니다.

인수
없음

반환값
없음

dom.stripTag()

지원 버전
Dreamweaver 3

설명
현재 선택 영역 주위의 태그를 제거하고 내용은 그대로 둡니다. 선택 영역에 태그가 둘 이상 있거나 하나도 없으면 Dreamweaver에서 오류를 보고합니다.

인수
없음

반환값
없음

dom.wrapTag()

지원 버전
Dreamweaver 3

설명
현재 선택 영역을 지정한 태그로 둘러쌉니다. 선택 영역에 있는 태그의 짝이 맞지 않는 경우 Dreamweaver에서 오류를 보고합니다.

인수

startTag, {bAlwaysBalance}, {bMakeLegal}

- startTag 인수는 열기 태그와 관련된 소스입니다.
- bAlwaysBalance 인수는 선택 영역을 둘러싸기 전에 선택 영역에서 태그의 짝을 맞추지 여부를 나타내는 부울 값입니다. 이 인수는 선택 사항입니다.
- bMakeLegal 인수는 둘러싸기 결과가 유효한 HTML인지를 확인할지 여부를 나타내는 부울 값입니다. 이 인수는 선택 사항입니다.

반환값

없음

예제

다음 코드에서는 현재 선택 영역을 링크로 둘러쌉니다.

```
var theDOM = dw.getDocumentDOM();
var theSel = theDOM.getSelectedNode();
if (theSel.nodeType == Node.TEXT_NODE) {
    theDOM.wrapTag('<a href="foo.html">');
}
```

dreamweaver.showQuickTagEditor()**지원 버전**

Dreamweaver 3

설명

현재 선택 영역에 대해 킥 태그 편집기를 표시합니다.

인수

{nearWhat}, {mode}

- 선택적인 **nearWhat** 인수는 "selection" 또는 "tag selector"여야 합니다. 이 인수를 생략하면 기본값은 "selection"입니다.
- 선택적인 **mode** 인수는 "default", "wrap", "insert" 또는 "edit"여야 합니다. **mode**가 "default"이거나 생략되면 Dreamweaver에서는 추론 기법을 통해 현재 선택 영역에 사용할 모드를 결정합니다. **nearWhat**이 "tag selector"이면 **mode** 인수가 무시됩니다.

반환값

없음

코드 뷰 함수

코드 뷰 함수에는 문서 소스 코드 편집과 관련된 작업이 포함되며, 디자인 뷰에 영향을 주는 작업도 포함됩니다. 이 단원의 함수를 사용하면 분할된 문서 뷰 내의 코드 뷰 또는 코드 관리자 윈도우에 탐색 컨트롤을 추가할 수 있습니다.

dom.formatRange()

지원 버전

Dreamweaver MX

설명

[환경 설정] > [코드 포맷] 대화 상자의 설정에 따라 [코드] 뷰에서 지정된 범위의 문자에 Dreamweaver 자동 구문 서식을 적용합니다.

인수

startOffset, endOffset

- **startOffset** 인수는 지정된 범위의 시작 부분을 문서 시작 부분부터의 오프셋으로 나타내는 정수입니다.
- **endOffset** 인수는 지정된 범위의 끝 부분을 문서 시작 부분부터의 오프셋으로 나타내는 정수입니다.

반환값

없음

dom.formatSelection()

지원 버전

Dreamweaver MX

설명

[환경 설정] > [코드 포맷] 대화 상자의 설정에 따라 선택한 내용에 Dreamweaver 자동 구문 서식을 적용합니다. 이것은 [명령] > [선택 영역에 소스 서식 적용] 옵션을 선택하는 것과 같습니다.

인수

없음

반환값

없음

dom.getShowNoscript()

지원 버전

Dreamweaver MX

설명

[보기] > [노스크립트 내용] 메뉴 옵션에서 noscript 내용 옵션의 현재 상태를 가져옵니다. 기본적으로 noscript 태그는 선택에 따라 브라우저에서 렌더링될 수 있는 페이지 스크립트 내용과 렌더링될 수 없는 페이지 스크립트 내용을 식별합니다.

인수

없음

반환값

부울 값을 반환합니다. noscript 태그 내용이 현재 렌더링되었으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.getAutoValidationCount()**지원 버전**

Dreamweaver MX 2004

설명

문서의 최근 자동 유효성 검사(인라인 유효성 검사라고도 함)에 대한 오류, 경고 및 정보 메시지의 수를 가져옵니다. 현재는 자동 유효성 검사를 수행하는 동안 대상 브라우저 확인만 수행됩니다(246페이지의 “[dom.runValidation\(\)](#)” 참조).

참고: 이 함수는 문서의 결과 윈도우에 현재 표시된 결과만 반환합니다. 최신 개수를 표시하려면 이 함수를 호출하기 전에 [dom.runValidation\(\)](#)을 호출합니다.

인수

없음

반환값

다음 속성이 있는 객체를 반환합니다.

- numError 속성: 오류의 수
- numWarning 속성: 경고의 수
- numInfo 속성: 정보 메시지의 수

예제

```
theDom = dw.getDocumentDOM();
theDom.runValidation();
theDom.getAutoValidationCount();
```

dom.isDesignViewUpdated()**지원 버전**

Dreamweaver 4

설명

유효한 문서 상태가 필요한 Dreamweaver 작업에 대해 [디자인] 뷰와 [텍스트] 뷰 내용이 동기화되었는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 디자인 뷰(WYSIWYG)가 텍스트 뷰의 텍스트와 동기화되었으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.isSelectionValid()

지원 버전

Dreamweaver 4

설명

선택 영역이 유효한지 여부를 확인합니다. 즉, 선택 영역이 [디자인] 뷰와 현재 동기화되어 있는지, 아니면 작업이 발생하기 전에 선택 영역을 이동해야 하는지를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 선택 영역이 유효한 코드이면 **true**를 반환하고 선택 영역이 업데이트되지 않아 문서가 동기화되지 않았으면 **false**를 반환합니다.

dom.setShowNoscript()

지원 버전

Dreamweaver MX

설명

noscript 내용 옵션을 설정하거나 해제합니다. [보기] > [노스크립트 내용] 옵션을 선택하는 것과 동일한 기능을 합니다. 기본적으로 noscript 태그는 선택에 따라 브라우저에서 렌더링될 수 있는 페이지 스크립트 내용과 렌더링될 수 없는 페이지 스크립트 내용을 식별합니다.

인수

{bShowNoscript}

- 선택적인 **bShowNoscript** 인수는 noscript 태그 내용을 렌더링할지 여부를 나타내는 부울 값입니다. noscript 태그 내용이 렌더링되면 **true**이고, 그렇지 않으면 **false**입니다.

반환값

없음

dom.source.arrowDown()

지원 버전

Dreamweaver 4

설명

삽입점을 행 단위로 [코드] 뷰 문서 아래로 이동합니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 행 단위로 확장합니다.

인수**{nTimes}, {bShiftIsDown}**

- 선택적인 **nTimes** 인수는 삽입점을 이동할 행 수입니다. **nTimes**를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.arrowLeft()**지원 버전**

Dreamweaver 4

설명

삽입점을 [코드] 뷰의 현재 행에서 왼쪽으로 이동합니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 왼쪽으로 확장합니다.

인수**{nTimes}, {bShiftIsDown}**

- 선택적인 **nTimes** 인수는 삽입점을 이동할 문자 수입니다. **nTimes**를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.arrowRight()**지원 버전**

Dreamweaver 4

설명

삽입점을 [코드] 뷰의 현재 행에서 오른쪽으로 이동합니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 오른쪽으로 확장합니다.

인수**{nTimes}, {bShiftIsDown}**

- 선택적인 **nTimes** 인수는 삽입점을 이동할 문자 수입니다. **nTimes**를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 것이고, 그렇지 않으면 선택되지 않은 것입니다.

반환값

없음

dom.source.arrowUp()

지원 버전

Dreamweaver 4

설명

삽입점을 행 단위로 [코드] 뷰 문서 위로 이동합니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 행 단위로 확장합니다.

인수

{nTimes}, {bShiftIsDown}

- **nTimes** 인수는 삽입점을 이동할 행 수입니다. **nTimes**를 생략하면 기본값은 1입니다.
- **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.balanceBracesTextView()

지원 버전

Dreamweaver 4

설명

이 함수는 괄호의 짝을 맞출 수 있는 [코드] 뷰 확장입니다. `dom.source.balanceBracesTextView()`를 호출하면 현재 강조 표시된 선택 영역 또는 삽입점을 확장할 수 있습니다. 확장은 둘러싸는 괄호 명령문의 시작부터 끝까지 이루어집니다. 이 작업은 [, {, ,, () 등의 문자와 같이 짝을 맞추기 위한 것입니다. 이후의 호출은 구두점 중첩의 단계까지 선택 영역을 확장하게 됩니다.

인수

없음

반환값

없음

dom.source.doCodeNavItem()

지원 버전

Dreamweaver 4

설명

이 함수는 코드 탐색기를 로드하고 현재 선택 영역에 대한 대상으로 채웁니다. 그러나 자체적으로 탐색하거나 관련 파일을 열지는 않습니다.

인수

없음

반환값

부울 값을 반환합니다. 코드 탐색기가 열리면 **true**를 반환하고 현재 선택 영역에 탐색 대상이 없기 때문에 코드 탐색기를 열 수 없으면 **false**를 반환합니다.

dom.source.endOfDocument()**지원 버전**

Dreamweaver 4

설명

삽입점을 현재 [코드] 뷰 문서의 마지막에 배치합니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 문서의 마지막까지 확장합니다.

인수

bShiftIsDown

- **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 **true**이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.endOfLine()**지원 버전**

Dreamweaver 4

설명

삽입점을 현재 행의 마지막에 둡니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 현재 행의 마지막까지 확장합니다.

인수

bShiftIsDown

- **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 **true**이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.endPage()**지원 버전**

Dreamweaver 4

설명

삽입점을 현재 페이지의 마지막으로 이동하거나 삽입점이 이미 페이지의 마지막에 있는 경우 다음 페이지의 마지막으로 이동합니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 페이지 단위로 확장합니다.

인수

{nTimes}, {bShiftIsDown}

- 선택적인 **nTimes** 인수는 삽입점을 이동할 페이지 수입니다. **nTimes**를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.getCurrentLines()**지원 버전**

Dreamweaver 4

설명

문서의 시작 부분에서부터 지정된 오프셋 위치의 행 번호를 반환합니다.

인수

없음

반환값

현재 선택 영역의 행 번호를 반환합니다.

dom.source.getSelection()**설명**

현재 문서의 선택 영역을 가져옵니다. 이것은 문서 [코드] 뷰까지의 문자 오프셋으로 표시됩니다.

인수

없음

반환값

소스 문서 시작 부분부터의 오프셋을 나타내는 정수의 쌍을 반환합니다. 첫 번째 정수는 선택 부분의 시작이고, 두 번째 정수는 선택 부분의 끝입니다. 두 숫자가 같은 경우 선택 영역은 삽입점이 됩니다. 소스에 선택 영역이 없는 경우 두 숫자는 모두 -1입니다.

dom.source.getLineFromOffset()**지원 버전**

Dreamweaver MX

설명

소스 문서에 오프셋을 적용합니다.

인수

없음

반환값

관련된 행의 번호를 반환하거나, 오프셋이 음수이거나 파일의 끝을 지나는 경우에는 -1을 반환합니다.

dom.source.getText()**지원 버전**

Dreamweaver 4

설명

지정된 오프셋 사이에 있는 소스의 텍스트 문자열을 반환합니다.

인수

startOffset, endOffset

- **startOffset** 인수는 문서 시작 부분부터의 오프셋을 나타내는 정수입니다.
- **endOffset** 인수는 문서의 끝을 나타내는 정수입니다.

반환값

start와 **end** 사이에 있는 소스 코드의 텍스트를 나타내는 문자열을 반환합니다.

dom.source.getValidationErrorsForOffset()**지원 버전**

Dreamweaver MX 2004

설명

지정된 오프셋에 있는 유효성 검사 오류 목록을 반환하거나 오프셋에서부터 다음 오류를 검색합니다. 오류가 검색되지 않으면 null이 반환됩니다.

인수

offset, {searchDirection}

- **offset** 인수는 함수에서 오류를 반환할 코드의 오프셋을 지정하는 숫자입니다.
- 선택적인 **searchDirection** 인수는 "empty", "forward" 또는 "back"을 지정하는 문자열입니다. 지정된 경우 함수는 지정된 오프셋의 앞 또는 뒤로 오류가 있는 다음 문자를 검색하여 반환합니다. 지정되지 않은 경우에는 단순히 지정된 오프셋에서 오류가 있는지 검사합니다.

반환값

객체 배열 또는 null 값을 반환합니다. 배열의 각 객체는 다음 속성을 갖습니다.

- **message** 객체는 오류 메시지를 포함하는 문자열입니다.
- **floatName** 객체는 결과 윈도우의 이름을 포함하는 문자열입니다. **showResults()** 또는 **setFloaterVisibility()** 함수에 이 값을 전달할 수 있습니다.
- **floatIndex** 객체는 플로터 결과 목록에 있는 항목의 인덱스입니다.
- **start** 객체는 밑줄로 표시된 코드의 열기 인덱스입니다.
- **end** 객체는 밑줄로 표시된 코드의 닫기 인덱스입니다.

참고: 반환된 플로터 인덱스는 문서가 열리거나 닫힐 때 자주 변경될 수 있으므로 저장할 수 없습니다.

예제

다음 예제에서는 `getValidationErrorsForOffset()`을 호출하여 현재 선택 영역의 오프셋에서 오류가 있는지 검사합니다. 함수가 오류를 반환하면 코드에서는 `alert()` 함수를 호출하여 사용자에게 오류 메시지를 표시합니다.

```
var offset = dw.getDocumentDOM().source.getSelection()[0];
var errors = dw.getDocumentDOM().source.getValidationErrorsForOffset(offset);
if ( errors && errors.length > 0 )
    alert ( errors[0].message );
```

dom.source.indentTextview()

지원 버전

Dreamweaver 4

설명

선택된 [코드] 뷰 텍스트를 탭 한 개만큼 오른쪽으로 이동합니다.

인수

없음

반환값

없음

dom.source.insert()

지원 버전

Dreamweaver 4

설명

소스 파일의 시작 부분에서부터 지정된 오프셋에 있는 소스 코드에 지정된 문자열을 삽입합니다. 오프셋이 0보다 작으면 삽입되지 않고 `false`가 반환됩니다.

인수

`offset`, `string`

- **offset** 인수는 문자열을 삽입할 파일 시작 부분부터의 오프셋입니다.
- **string** 인수는 삽입할 문자열입니다.

반환값

부울 값, 성공적으로 실행되면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

dom.source.nextWord()

지원 버전

Dreamweaver 4

설명

[코드] 뷰에서 삽입점을 다음 단어의 처음으로 이동합니다. 이동할 단어 수가 지정될 수도 있습니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 오른쪽으로 확장합니다.

인수

{nTimes}, {bShiftIsDown}

- 선택적인 **nTimes** 인수는 삽입점을 이동할 단어 수입니다. **nTimes**를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.outdentTextview()**지원 버전**

Dreamweaver 4

설명

선택된 [코드] 뷰 텍스트를 탭 한 개만큼 왼쪽으로 이동합니다.

인수

없음

반환값

없음

dom.source.pageDown()**지원 버전**

Dreamweaver 4

설명

삽입점을 페이지 단위로 [코드] 뷰 문서 아래로 이동합니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 페이지 단위로 확장합니다.

인수

{nTimes}, {bShiftIsDown}

- 선택적인 **nTimes** 인수는 삽입점을 이동할 페이지 수입니다. **nTimes**를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.pageUp()

지원 버전

Dreamweaver 4

설명

삽입점을 페이지 단위로 [코드] 뷰 문서 위로 이동합니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 페이지 단위로 확장합니다.

인수

{nTimes}, {bShiftIsDown}

- 선택적인 **nTimes** 인수는 삽입점을 이동할 페이지 수입니다. **nTimes**를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.previousWord()

지원 버전

Dreamweaver 4

설명

[코드] 뷰에서 삽입점을 이전 단어의 처음으로 이동합니다. 이동할 단어 수가 지정될 수도 있습니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 왼쪽으로 확장합니다.

인수

{nTimes}, {bShiftIsDown}

- 선택적인 **nTimes** 인수는 삽입점을 이동할 단어 수입니다. **nTimes**를 생략하면 기본값은 1입니다.
- 선택적인 **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.replaceRange()

지원 버전

Dreamweaver 4

설명

startOffset과 **endOffset** 사이 범위의 소스 텍스트를 **string**으로 대체합니다. **startOffset**이 **endOffset**보다 크거나 두 오프셋 중 하나가 양의 정수가 아닌 경우에는 아무 작업도 수행하지 않고 **false**를 반환합니다. **endOffset**이 파일의 문자 수보다 클 경우 **startOffset**과 파일 끝 사이의 범위로 대체됩니다. **startOffset**과 **endOffset** 모두 파일의 문자 수보다 클 경우 텍스트를 파일의 끝에 삽입합니다.

인수

startOffset, endOffset, string

- **startOffset** 인수는 대체할 블록의 시작 부분을 나타내는 오프셋입니다.
- **endOffset** 인수는 대체할 블록의 끝 부분을 나타내는 오프셋입니다.
- **string** 인수는 삽입할 문자열입니다.

반환값

부울 값, 성공적으로 실행되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.source.scrollEndFile()**지원 버전**

Dreamweaver 4

설명

삽입점을 이동하지 않고 [코드] 뷰를 문서 파일의 맨 아래로 스크롤합니다.

인수

없음

반환값

없음

dom.source.scrollLineDown()**지원 버전**

Dreamweaver 4

설명

삽입점을 이동하지 않고 행 단위로 [코드] 뷰를 아래로 스크롤합니다.

인수

nTimes

- **nTimes** 인수는 스크롤할 행의 수입니다. **nTimes**를 생략하면 기본값은 1입니다.

반환값

없음

dom.source.scrollLineUp()

지원 버전

Dreamweaver 4

설명

삽입점을 이동하지 않고 행 단위로 [코드] 뷰를 위로 스크롤합니다.

인수

nTimes

- nTimes 인수는 스크롤할 행의 수입니다. nTimes를 생략하면 기본값은 1입니다.

반환값

없음

dom.source.scrollPageDown()

지원 버전

Dreamweaver 4

설명

삽입점을 이동하지 않고 페이지 단위로 [코드] 뷰를 아래로 스크롤합니다.

인수

nTimes

- nTimes 인수는 스크롤할 페이지 수입니다. nTimes를 생략하면 기본값은 1입니다.

반환값

없음

dom.source.scrollPageUp()

지원 버전

Dreamweaver 4

설명

삽입점을 이동하지 않고 페이지 단위로 [코드] 뷰를 위로 스크롤합니다.

인수

nTimes

- nTimes 인수는 스크롤할 페이지 수입니다. nTimes를 생략하면 기본값은 1입니다.

반환값

없음

dom.source.scrollTopFile()

지원 버전

Dreamweaver 4

설명

삽입점을 이동하지 않고 [코드] 뷰를 문서 파일의 맨 위로 스크롤합니다.

인수

없음

반환값

없음

dom.source.selectParentTag()

지원 버전

Dreamweaver 4

설명

이 함수는 태그의 짝을 맞출 수 있는 [코드] 뷰 확장입니다. dom.source.selectParentTag()를 호출하면 현재 강조 표시된 선택 영역 또는 삽입점을 이를 둘러싸는 열기 태그에서 닫기 태그까지 확장할 수 있습니다. 이 함수를 또 호출하면 선택 영역을 둘러싸는 추가 태그까지 확장하며 이 과정은 더 이상의 포함 태그가 없을 때까지 계속됩니다.

인수

없음

반환값

없음

dom.source.setCurrentLine()

지원 버전

Dreamweaver 4

설명

지정된 행의 시작 부분에 삽입점을 놓습니다. lineNumber 인수가 양의 정수가 아닌 경우 함수는 아무 작업도 수행하지 않고 false를 반환합니다. lineNumber가 소스의 행 수보다 큰 경우에는 마지막 행의 시작 부분에 삽입점을 놓습니다.

인수

lineNumber

- lineNumber 인수는 시작 부분에 삽입점을 놓을 행입니다.

반환값

부울 값, 성공적으로 실행되면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.source.startOfDocument()

지원 버전

Dreamweaver 4

설명

삽입점을 [코드] 뷰 문서의 처음에 놓습니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 문서의 처음까지 확장합니다.

인수

bShiftIsDown

- **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.startOfLine()

지원 버전

Dreamweaver 4

설명

삽입점을 현재 행의 처음에 놓습니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 현재 행의 처음으로 확장합니다.

인수

bShiftIsDown

- **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.topPage()

지원 버전

Dreamweaver 4

설명

삽입점을 현재 페이지의 맨 위로 이동하거나 삽입점이 이미 페이지의 맨 위에 있는 경우 이전 페이지의 맨 위로 이동합니다. 내용이 이미 선택되어 있는 경우, 이 함수는 선택 영역을 페이지 단위로 확장합니다.

인수

{nTimes}, {bShiftIsDown}

- 선택적인 **nTimes** 인수는 삽입점을 이동할 페이지 수입니다. **nTimes**를 생략하면 기본값은 1입니다.

- 선택적인 **bShiftIsDown** 인수는 내용이 선택되는지 여부를 나타내는 부울 값입니다. **bShiftIsDown**이 true이면 내용이 선택된 경우입니다.

반환값

없음

dom.source.wrapSelection()

지원 버전

Dreamweaver 4

설명

현재 선택 영역 앞에 **startTag**의 텍스트를, 현재 선택 영역 뒤에 **endTag**의 텍스트를 삽입합니다. 그런 다음 삽입된 태그를 포함하여 태그 사이의 전체 범위를 선택합니다. 현재 선택 영역이 삽입점인 경우에는 **startTag**와 **endTag** 사이에 삽입점을 놓습니다. **startTag**와 **endTag**는 태그일 필요는 없으며, 임의의 텍스트가 될 수 있습니다.

인수

startTag, **endTag**

- startTag** 인수는 선택 영역의 처음에 삽입할 텍스트입니다.
- endTag** 인수는 선택 영역의 끝에 삽입할 텍스트입니다.

반환값

없음

dom.synchronizeDocument()

지원 버전

Dreamweaver 4

설명

[디자인] 및 [코드] 뷰를 동기화합니다.

인수

없음

반환값

없음

태그 편집기 및 태그 라이브러리 함수

태그 편집기를 사용하면 새 태그를 삽입하고, 기존 태그를 편집하고, 태그에 대한 참조 정보에 액세스할 수 있습니다. 태그 선택기를 사용하면 자주 사용하는 태그를 쉽게 선택할 수 있도록 태그를 구성할 수 있습니다. Dreamweaver의 태그 라이브러리에는 표준 기반 마크업 언어와 가장 널리 사용되는 태그 기반 스크립팅 언어에 사용하는 태그에 대한 정보가 저장됩니다. Extension에서 태그 편집기 및 태그 라이브러리에 액세스하여 작업할 필요가 있는 경우 JavaScript 태그 편집기, 태그 선택기 및 태그 라이브러리 함수를 사용하면 됩니다.

dom.getTagSelectorTag()

지원 버전

Dreamweaver MX

설명

이 함수는 문서 윈도우의 아래쪽에 있는 태그 선택기 막대에 현재 선택되어 있는 태그의 DOM 노드를 가져옵니다.

인수

없음

반환값

현재 선택되어 있는 태그의 DOM 노드를 반환하거나, 선택한 태그가 없는 경우에는 null을 반환합니다.

dreamweaver.popupInsertTagDialog()

지원 버전

Dreamweaver MX

설명

이 함수는 VTM 파일을 검사하여 태그에 태그 편집기가 정의되어 있는지 확인합니다. 정의되어 있는 경우 해당 태그의 편집기가 나타나고 시작 태그가 입력됩니다. 그렇지 않은 경우에는 시작 태그가 수정 없이 사용자의 문서에 삽입됩니다.

인수

start_tag_string

다음 중 한 가지 유형의 초기 값이 들어 있는 시작 태그 문자열입니다.

- <input>과 같은 태그
- <input type='text'>와 같이 속성이 있는 태그
- <%= %>와 같은 지시문

반환값

부울 값을 반환합니다. 문서에 삽입된 항목이 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.popupEditTagDialog()

지원 버전

Dreamweaver MX

설명

선택한 태그가 있으면 이 함수는 태그를 편집할 수 있도록 해당 태그에 대한 태그 편집기를 엽니다.

인수

없음

반환값

없음

활성자

445페이지의 “[dreamweaver.canPopupEditTagDialog\(\)](#)”를 참조하십시오.

dreamweaver.showTagChooser()

지원 버전

Dreamweaver MX

설명

이 함수는 [태그 선택기] 대화 상자를 표시하고 맨 앞으로 가져온 다음 포커스를 해당 대화 상자에 둡니다.

인수

없음

반환값

없음

dreamweaver.showTagLibraryEditor()

지원 버전

Dreamweaver MX

설명

이 함수는 태그 라이브러리 편집기를 엽니다.

인수

없음

반환값

없음

dreamweaver.tagLibrary.getTagLibraryDOM()

지원 버전

Dreamweaver MX

설명

filename.vtm 파일의 URL을 지정하면 이 함수는 내용을 편집할 수 있도록 해당 파일의 DOM을 반환합니다. 이 함수는 태그 라이브러리 편집기가 활성화된 상태에서만 호출해야 합니다.

인수

fileURL

- **fileURL** 인수는 Configuration/Tag Libraries 폴더에 상대적인 **filename.vtm** 파일의 URL입니다(예: "HTML/img.vtm").

반환값

TagLibraries 폴더 내의 새 파일 또는 기존 파일을 가리키는 DOM 포인터를 반환합니다.

dreamweaver.tagLibrary.getSelectedLibrary()

지원 버전

Dreamweaver MX

설명

태그 라이브러리 편집기에서 라이브러리 노드를 선택한 경우 이 함수는 라이브러리 이름을 가져옵니다.

인수

없음

반환값

태그 라이브러리 편집기에서 현재 선택되어 있는 라이브러리의 이름 문자열. 선택된 라이브러리가 없는 경우 빈 문자열을 반환합니다.

dreamweaver.tagLibrary.getSelectedTag()

지원 버전

Dreamweaver MX

설명

속성 노드가 현재 선택되어 있는 경우 이 함수는 속성이 들어 있는 태그의 이름을 가져옵니다.

인수

없음

반환값

태그 라이브러리 편집기에서 현재 선택되어 있는 태그의 이름 문자열을 반환하거나, 선택된 태그가 없는 경우에는 빈 문자열을 반환합니다.

dreamweaver.tagLibrary.importDTDOrSchema()**지원 버전**

Dreamweaver MX

설명

이 함수는 원격 서버의 DTD나 스키마 파일을 태그 라이브러리로 가져옵니다.

인수

fileURL, Prefix

- **fileURL** 인수는 로컬 URL 형식의 DTD 또는 스키마 파일 경로입니다.
- **Prefix** 인수는 이 태그 라이브러리에 있는 모든 태그에 추가될 접두사 문자열입니다.

반환값

가져온 태그 라이브러리의 이름을 반환합니다.

dreamweaver.tagLibrary.getImportedTagList()**지원 버전**

Dreamweaver MX

설명

이 함수는 가져온 태그 라이브러리의 tagInfo 객체 목록을 생성합니다.

인수

libname

- **libname** 인수는 가져온 태그 라이브러리의 이름입니다.

반환값

tagInfo 객체의 배열을 반환합니다.

taginfo 객체에는 태그 라이브러리에 포함된 단일 태그에 대한 정보가 들어 있습니다. tagInfo 객체에서는 다음 속성이 정의됩니다.

- **tagName** 속성은 문자열입니다.
- **attributes** 속성은 문자열 배열입니다. 각 문자열은 이 태그에 대해 정의된 속성의 이름입니다.

예제:

다음 예제에서는 dw.tagLibrary.getImportedTagList() 함수를 사용하면 libName 라이브러리에서 태그 배열을 가져올 수 있다는 것을 보여 줍니다.

```
// "fileURL" and "prefix" have been entered by the user.  
// tell the Tag Library to Import the DTD/Schema  
var libName = dw.tagLibrary.importDTDOrSchema(fileURL, prefix);  
  
// get the array of tags for this library  
// this is the TagInfo object  
var tagArray = dw.tagLibrary.getImportedTagList(libName);  
  
// now I have an array of tagInfo objects.  
// I can get info out of them. This gets info out of the first one.  
// note: this assumes there is at least one TagInfo in the array.  
var firstTagName = tagArray[0].name;  
var firstTagAttributes = tagArray[0].attributes;  
// note that firstTagAttributes is an array of attributes.
```

19장: 활성화자

Adobe® Dreamweaver® CS4 활성화자 함수는 다른 함수가 현재 컨텍스트에서 특정 작업을 수행할 수 있는지 여부를 확인합니다. 함수 사양은 각 함수가 true 값을 반환하는 일반적 상황을 설명합니다. 하지만 이 설명은 전체를 포괄하지 않으며 함수가 false 값을 반환하는 일부 경우가 제외될 수도 있습니다.

활성화자 함수

JavaScript API의 활성화자 함수에는 다음과 같은 함수가 포함됩니다.

dom.canAlign()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [왼쪽 정렬], [오른쪽 정렬], [위쪽 정렬] 및 [아래쪽 정렬]을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

두 개 이상의 레이어 또는 핫스팟이 선택되어 있는지 여부를 나타내는 부울 값을 반환합니다.

dom.canApplyTemplate()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [페이지에 적용] 작업을 수행할 수 있는지 여부를 확인합니다. 이 함수는 활성 문서에만 사용할 수 있습니다.

인수

없음

반환값

문서가 라이브러리 항목 또는 템플릿이 아닌지 여부와 선택 영역이 NOFRAMES 태그 내에 있지 않은지 여부를 나타내는 부울 값을 반환합니다.

dom.canArrange()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [앞으로 가져오기] 또는 [뒤로 보내기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

핫스팟이 선택되어 있는지 여부를 나타내는 부울 값을 반환합니다.

dom.canClipCopyText()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [텍스트로 복사] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역의 열기 및 닫기 오프셋이 다르면 **true**를 반환하고 그렇지 않으면 **false**를 반환하여 선택된 영역이 없음을 나타냅니다.

dom.canClipPaste()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [붙여넣기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. Dreamweaver에 붙여 넣을 수 있는 내용이 클립보드에 들어 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.canClipPasteText()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [텍스트로 붙여넣기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. Dreamweaver에 텍스트로 붙여 넣을 수 있는 내용이 클립보드에 들어 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canConvertLayersToTable()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [레이어를 표로 변환] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 문서의 BODY 섹션에 있는 모든 내용이 레이어에 들어 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canConvertTablesToLayers()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [표를 레이어로 변환] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 문서의 BODY 섹션에 있는 모든 내용이 표에 들어 있고 문서가 템플릿 기반이 아니면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canDecreaseColspan()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [가로 셀 나누기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 셀에 COLSPAN 속성이 있고 해당 속성 값이 2보다 크거나 같으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canDecreaseRowspan()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [세로 셀 나누기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 셀에 ROWSPAN 속성이 있고 해당 속성 값이 2보다 크거나 같으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canDeleteTableColumn()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [열 삭제] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 삽입점이 셀 안에 있거나 셀 또는 열이 선택되어 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canDeleteTableRow()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [행 삭제] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 삽입점이 셀 안에 있거나 셀 또는 행이 선택되어 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

site.canEditColumns()

설명

사이트가 존재하는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 사이트가 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canEditNoFramesContent()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [비프레임 내용 편집] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 문서가 프레임세트이거나 프레임세트 안에 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canIncreaseColspan()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [가로 셀 합치기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 셀 오른쪽에 셀이 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canIncreaseRowspan()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [세로 셀 합치기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 셀 아래에 셀이 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canInsertTableColumns()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [열 삽입] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역이 표 안에 있으면 true를 반환하고 선택 영역이 전체 표이거나 표 안에 없으면 false를 반환합니다.

dom.canInsertTableRows()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [행 삽입] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역이 표 안에 있으면 **true**를 반환하고 선택 영역이 전체 표이거나 표 안에 없으면 **false**를 반환합니다.

dom.canMakeNewEditableRegion()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [새 편집 가능 영역] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 문서가 템플릿(DWT) 파일이면 **true**를 반환합니다.

dom.canMarkSelectionAsEditable()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [선택 내용을 편집 가능 영역으로 표시] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역이 있고 현재 문서가 DWT 파일이면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.canMergeTableCells()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [셀 병합] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역이 표 셀의 인접 그룹이면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.canPlayPlugin()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [재생] 작업을 수행할 수 있는지 여부를 확인합니다. 이 함수는 활성 문서에만 사용할 수 있습니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역을 플러그인으로 재생할 수 있으면 **true**를 반환합니다.

dom.canRedo()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [다시 실행] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 다시 실행할 단계가 남아 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.canRemoveEditableRegion()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [편집 가능 영역 제거] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 문서가 템플릿이면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.canSelectTable()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [표 선택] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 삽입점 또는 선택 영역이 표 안에 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.canSetLinkHref()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 현재 선택 영역 주위의 링크를 변경하거나, 필요한 경우 새로 만들 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역이 이미지, 텍스트이거나 삽입점이 링크 안에 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다. 텍스트 선택 영역은 텍스트 속성 관리자가 표시될 선택 영역으로 정의됩니다.

dom.canShowListPropertiesDialog()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [목록 속성] 대화 상자를 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역이 LI 태그 안에 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dom.canSplitFrame()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [[왼쪽 | 오른쪽 | 위쪽 | 아래쪽]으로 프레임 분할] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역이 프레임 안에 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canSplitTableCell()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [셀 분할] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 삽입점이 표 셀 안에 있거나 선택 영역이 표 셀이면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canStopPlugin()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [중단] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역이 플러그인을 통해 현재 재생되고 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.canUndo()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [실행 취소] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 실행 취소할 단계가 남아 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dom.hasTracingImage()

지원 버전

Dreamweaver 3

설명

문서에 바탕 이미지가 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 문서에 바탕 이미지가 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.assetPalette.canEdit()

지원 버전

Dreamweaver 4

설명

[에셋] 패널의 메뉴 항목을 편집할 수 있도록 활성화합니다.

인수

없음

반환값

부울 값을 반환합니다. 에셋을 편집할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다. [사이트] 목록의 색상 및 URL에 대해서는 false를 반환하고 [즐거찾기] 목록의 색상 및 URL을 여러 개 선택한 경우에도 false를 반환합니다.

dreamweaver.assetPalette.canInsertOrApply()

지원 버전

Dreamweaver 4

설명

선택된 요소를 삽입 또는 적용할 수 있는지 여부를 확인합니다. **true** 또는 **false** 값을 반환하여 삽입 또는 적용할 메뉴 항목을 활성화하거나 비활성화합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택된 요소를 삽입 또는 적용할 수 있으면 **true**를 반환하고 현재 페이지가 템플릿이고 현재 범주가 [템플릿]이면 **false**를 반환합니다. 이 함수는 열려 있는 문서가 없거나, 문서에서 라이브러리 항목이 선택되어 있고 현재 범주가 [라이브러리]인 경우에도 **false**를 반환합니다.

dreamweaver.canClipCopy()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [복사] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 클립보드로 복사할 수 있는 내용이 선택되어 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.canClipCut()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [잘라내기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 클립보드로 잘라낼 수 있는 내용이 선택되어 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.canClipPaste()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [붙여넣기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 문서 또는 [사이트] 패널의 활성 윈도우(Macintosh에서는 부동 패널이나 대화 상자의 텍스트 필드)로 붙여 넣을 수 있는 내용이 클립보드에 들어 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.canDeleteSelection()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 현재 선택 영역을 삭제할 수 있는지 여부를 확인합니다. 포커스가 있는 윈도우에 따라 문서 윈도우 또는 사이트 패널(Macintosh에서는 대화 상자 또는 부동 패널의 텍스트 필드)에서 삭제가 수행될 수 있습니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역의 열기 및 닫기 오프셋이 다르면 선택 영역이 있음을 나타내는 true를 반환하고 오프셋이 같으면 삽입점만 있음을 나타내는 false를 반환합니다.

dreamweaver.canExportTemplateDataAsXML()

지원 버전

Dreamweaver MX

설명

Dreamweaver에서 현재 문서를 XML로 내보낼 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 문서에서 내보내기를 수행할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

예제

다음 예제에서는 `dw.canExportTemplateDataAsXML()`을 호출하여 Dreamweaver에서 현재 문서를 XML로 내보낼 수 있는지 여부를 확인한 다음 `true`가 반환되면 `dw.ExportTemplateDataAsXML()`을 호출하여 내보냅니다.

```
if (dreamweaver.canExportTemplateDataAsXML())  
{  
    dreamweaver.exportTemplateDataAsXML("file:///c:/dw_temps/mytemplate.txt")  
}
```

dreamweaver.canFindNext()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [다음 찾기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 검색 패턴이 이미 만들어져 있으면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

dreamweaver.canFitSelection()

지원 버전

Dreamweaver 8

설명

활성 [디자인] 뷰에 선택 영역이 있는지, 즉, `fitSelection()`을 호출할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 활성 [디자인] 뷰에 선택 영역이 있으면 `true`를 반환하고 그렇지 않으면 `false`를 반환합니다.

dreamweaver.canOpenInFrame()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [프레임에서 열기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역 또는 삽입점가 프레임 안에 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.canPasteSpecial()**지원 버전**

Dreamweaver 8

설명

Dreamweaver에서 [선택하여 붙여넣기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 클립보드에 텍스트, HTML 또는 Dreamweaver HTML이 있고 포커스가 [코드] 뷰, [디자인] 뷰 또는 코드 관리자에 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.canPlayRecordedCommand()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [기록된 명령 실행] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 활성 문서가 있으며 이전에 기록된 명령을 실행할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.canPopupEditTagDialog()**지원 버전**

Dreamweaver MX

설명

현재 선택 영역이 태그인지 여부와 [태그 편집] 메뉴 항목의 활성화 여부를 확인합니다.

인수

없음

반환값

현재 선택된 태그의 이름을 반환하거나, 선택된 태그가 없는 경우에는 null 값을 반환합니다.

dreamweaver.canRedo()**지원 버전**

Dreamweaver 3

설명

Dreamweaver가 현재 컨텍스트에서 [다시 실행] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

임의의 작업을 실행 취소할 수 있는지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.canRevertDocument()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 마지막에 저장한 상태로 [복귀] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

documentObject

- documentObject 인수는 문서의 DOM 트리 루트에 있는 객체(dreamweaver.getDocumentDOM() 함수가 반환하는 값)입니다.

반환값

문서가 저장되지 않은 상태이고 문서의 저장된 버전이 로컬 드라이브에 존재하는지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.canSaveAll()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [모두 저장] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

저장되지 않은 문서가 하나 이상 열려 있는지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.canSaveDocument()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 지정한 문서에 대해 [저장] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

documentObject

- documentObject 인수는 문서의 DOM 루트(dreamweaver.getDocumentDOM() 함수가 반환하는 값)입니다.

반환값

문서에 저장되지 않은 변경 내용이 있는지 나타내는 부울 값을 반환합니다.

dreamweaver.canSaveDocumentAsTemplate()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 지정한 문서에 대해 [템플릿으로 저장] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

documentObject

- documentObject 인수는 문서의 DOM 루트(dreamweaver.getDocumentDOM() 함수가 반환하는 값)입니다.

반환값

문서를 템플릿으로 저장할 수 있는지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.canSaveFrameset()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 지정한 문서에 대해 [프레임세트 저장] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

documentObject

- documentObject 인수는 문서의 DOM 루트(dreamweaver.getDocumentDOM() 함수가 반환하는 값)입니다.

반환값

문서가 저장되지 않은 변경 내용이 있는 프레임셋인지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.canSaveFramesetAs()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 지정한 문서에 대해 [다른 이름으로 프레임셋 저장] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

documentObject

- **documentObject** 인수는 문서의 DOM 루트(`dreamweaver.getDocumentDOM()` 함수가 반환하는 값)입니다.

반환값

문서가 프레임셋인지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.canSelectAll()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [모두 선택] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

[모두 선택] 작업을 수행할 수 있는지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.canShowFindDialog()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [찾기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값, [사이트] 패널 또는 [문서] 윈도우가 열려 있으면 **true**를 반환합니다. 이 함수는 선택 영역이 HEAD 섹션에 있으면 **false**를 반환합니다.

dreamweaver.canUndo()**지원 버전**

Dreamweaver 3

설명

Dreamweaver가 현재 컨텍스트에서 [실행 취소] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

임의의 작업을 실행 취소할 수 있는지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.canZoom()**지원 버전**

Dreamweaver 8

설명

활성화된 [디자인] 뷰가 있는지, 즉 기본 확대/축소 명령을 적용할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 활성 [디자인] 뷰가 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.cssRuleTracker.canEditSelectedRule()**지원 버전**

Dreamweaver MX 2004

설명

선택된 규칙에 속성 격자 편집기를 적용할 수 있는지 여부를 확인합니다. 속성 격자에는 잠긴 파일의 규칙이 표시될 수 있으므로 **true**가 반환된다고 해서 반드시 규칙을 수정할 수 있는 것은 아닙니다.

인수

없음

반환값

부울 값을 반환합니다. 선택된 규칙에 속성 격자 편집기를 적용할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

예제

다음 코드에서는 선택된 규칙을 편집할 수 있도록 허용하기 전에 활성 자 함수의 값이 **true**로 설정되었는지 여부를 확인합니다.

```
if(dw.cssRuleTracker.canEditSelectedRule()){
    dw.cssRuleTracker.editSelectedRule();
}
```

dreamweaver.cssStylePalette.canApplySelectedStyle()**지원 버전**

Dreamweaver MX

설명

선택된 스타일을 현재 활성 문서에 적용할 수 있는지 여부를 확인합니다.

인수

{pane}

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

부울 값을 반환합니다. 선택된 스타일에 클래스 선택기가 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.cssStylePalette.canDeleteSelectedStyle()**지원 버전**

Dreamweaver MX

설명

현재 선택 영역에서 선택된 스타일을 삭제할 수 있는지 여부를 확인합니다.

인수

{pane}

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

부울 값을 반환합니다. 선택 영역을 삭제할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.cssStylePalette.canDuplicateSelectedStyle()

지원 버전

Dreamweaver MX

설명

현재 활성 문서에서 선택된 스타일을 복제할 수 있는지 여부를 확인합니다.

인수

{pane}

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서의 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

부울 값을 반환합니다. 선택된 스타일을 복제할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.cssStylePalette.canEditSelectedStyle()

지원 버전

Dreamweaver MX

설명

현재 활성 문서에서 선택된 스타일을 편집할 수 있는지 여부를 확인합니다.

인수

{pane}

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서의 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

부울 값을 반환합니다. 선택된 스타일을 편집할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.cssStylePalette.canEditSelectedStyleInCodeview()

지원 버전

Dreamweaver MX

설명

현재 활성 문서에서 선택된 스타일을 [코드] 뷰에서 편집할 수 있는지 여부를 확인합니다.

인수**{pane}**

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

부울 값을 반환합니다. 선택된 스타일을 편집할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.cssStylePalette.canEditStyleSheet()**지원 버전**

Dreamweaver MX

설명

현재 선택 영역에 편집이 가능한 스타일 시트 요소가 들어 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택 영역이 스타일 시트 노드이거나 스타일 시트 노드 내의 스타일 정의이고 스타일 시트가 숨겨져 있지도 않고 이 문서도 아니면 true를 반환하고 선택 영역이 숨겨져 있거나 이 문서에 속하면 false를 반환합니다.

dreamweaver.cssStylePalette.canRenameSelectedStyle()**지원 버전**

Dreamweaver MX

설명

현재 활성 문서에서 선택된 스타일의 이름을 변경할 수 있는지 여부를 확인합니다.

인수**{pane}**

- 선택적인 **pane** 인수는 이 함수를 적용할 [스타일] 패널의 구역을 지정하는 문자열입니다. 사용할 수 있는 값은 "All" 모드에서 스타일 목록인 "stylelist", "Current" 모드에서 관련된 해당 규칙 목록인 "cascade", "Current" 모드에서 현재 선택 사항에 대한 속성 목록인 "summary", "Current" 모드에서 속성의 편집 가능한 목록 또는 격자인 "ruleInspector" 등입니다. 기본값은 "stylelist"입니다.

반환값

부울 값을 반환합니다. 선택된 스타일의 이름을 변경할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.isRecording()

지원 버전

Dreamweaver 3

설명

현재 Dreamweaver에서 명령을 기록 중인지 여부를 보고합니다.

인수

없음

반환값

Dreamweaver에서 명령을 기록 중인지 여부를 나타내는 부울 값을 반환합니다.

dreamweaver.htmlStylePalette.canEditSelection()

지원 버전

Dreamweaver 3

설명

Dreamweaver가 [HTML 스타일] 패널에서 선택 영역을 편집, 삭제 또는 복제할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. Dreamweaver가 [HTML 스타일] 패널에서 선택 영역을 편집, 삭제 또는 복제할 수 있으면 **true**를 반환하고 선택된 스타일이 없거나 clear 스타일 중 하나가 선택되어 있으면 **false**를 반환합니다.

dreamweaver.resultsPalette.canClear()

지원 버전

Dreamweaver MX

설명

현재 포커스가 있는 [결과] 패널의 내용을 지울 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 내용을 지울 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.resultsPalette.canCopy()

지원 버전

Dreamweaver MX

설명

현재 [결과] 윈도우의 내용에 복사된 메시지를 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 내용을 표시할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.resultsPalette.canCut()

지원 버전

Dreamweaver MX

설명

현재 [결과] 윈도우의 내용에 [잘라내기] 메시지를 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 내용을 표시할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.resultsPalette.canPaste()

지원 버전

Dreamweaver MX

설명

현재 [결과] 윈도우의 내용에 [붙여넣기] 메시지를 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 내용을 표시할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.resultsPalette.canOpenInBrowser()

지원 버전

Dreamweaver MX

설명

현재 보고서를 브라우저에 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 내용을 표시할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.resultsPalette.canOpenInEditor()

지원 버전

Dreamweaver MX

설명

현재 보고서를 편집기에 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 내용을 표시할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.resultsPalette.canSave()

지원 버전

Dreamweaver MX

설명

현재 패널에 대해 [저장] 대화 상자를 열 수 있는지 여부를 확인합니다. 현재 [사이트 보고서], [대상 브라우저 확인], [유효성 검사] 및 [링크 확인기] 패널이 [저장] 대화 상자를 지원합니다.

인수

없음

반환값

부울 값을 반환합니다. [저장] 대화 상자를 표시할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.resultsPalette.canSelectAll()

지원 버전

Dreamweaver MX

설명

현재 포커스가 있는 윈도우에 [모두 선택] 메시지를 전송할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [모두 선택] 메시지를 전송할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.siteSyncDialog.canCompare()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 [비교] 컨텍스트 메뉴를 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [사이트 동기화] 대화 상자에 [비교] 컨텍스트 메뉴를 표시할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.siteSyncDialog.canMarkDelete()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 [삭제로 액션 변경] 컨텍스트 메뉴를 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [삭제로 액션 변경] 컨텍스트 메뉴를 표시할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.siteSyncDialog.canMarkGet()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 [가져오기로 액션 변경] 컨텍스트 메뉴를 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [가져오기로 액션 변경] 컨텍스트 메뉴를 표시할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.siteSyncDialog.canMarkIgnore()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 [무시로 액션 변경] 컨텍스트 메뉴를 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [무시로 액션 변경] 컨텍스트 메뉴를 표시할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.siteSyncDialog.canMarkPut()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 [올리기로 액션 변경] 컨텍스트 메뉴를 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [올리기로 액션 변경] 컨텍스트 메뉴를 표시할 수 있으면 true를 반환하고 그렇지 않으면 false를 반환합니다.

dreamweaver.siteSyncDialog.canMarkSynced()

지원 버전

Dreamweaver 8

설명

이 함수는 [사이트 동기화] 대화 상자에 [동기화로 액션 변경] 컨텍스트 메뉴를 표시할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [동기화로 액션 변경] 컨텍스트 메뉴를 표시할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.snippetpalette.canEditSnippet()

지원 버전

Dreamweaver MX

설명

현재 선택된 항목을 편집할 수 있는지 확인한 다음 편집할 메뉴 항목을 활성화 또는 비활성화할 수 있도록 **true** 또는 **false** 값을 반환합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 선택된 항목을 편집할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

dreamweaver.snippetpalette.canInsert()

지원 버전

Dreamweaver MX

설명

선택된 요소를 삽입 또는 적용할 수 있는지 확인한 다음 삽입 또는 적용할 메뉴 항목을 활성화하거나 비활성화할 수 있도록 **true** 또는 **false** 값을 반환합니다.

인수

없음

반환값

부울 값을 반환합니다. 선택된 요소를 삽입 또는 적용할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.browseDocument()

지원 버전

Dreamweaver 4

설명

선택된 모든 문서를 브라우저 윈도우에서 엽니다. [브라우저에서 미리 보기] 명령을 사용하는 것과 같습니다.

인수

browserName

- **browserName** 인수는 [브라우저에서 미리 보기] 환경 설정에서 정의한 브라우저 이름입니다. 이 인수가 생략되는 경우 기본값은 사용자의 기본 브라우저입니다.

반환값

없음

site.canCheckIn()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [체크 인] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

siteOrURL

- **siteOrURL** 인수는 site 키워드여야 합니다. 이는 함수가 단일 파일의 URL 또는 [사이트] 패널의 선택 영역에 적용되어야 함을 의미합니다.

반환값

부울 값을 반환합니다. 다음 조건이 모두 참이면 true를 반환하고 그렇지 않으면 false를 반환합니다.

- 원격 사이트가 정의되어 있습니다.
- 문서 윈도우에 포커스가 있는 경우 파일이 로컬 사이트에 저장되어 있으며, [사이트] 패널에 포커스가 있는 경우 하나 이상의 파일 또는 폴더가 선택되어 있습니다.
- 사이트에 대해 체크 인/체크 아웃 기능이 설정되어 있습니다.

site.canCheckOut()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 지정한 파일에 대해 [체크 아웃] 작업을 수행할 수 있는지 여부를 확인합니다.

인수**siteOrURL**

- **siteOrURL** 인수는 **site** 키워드여야 합니다. 이는 함수가 단일 파일의 **URL** 또는 **[사이트]** 패널의 선택 영역에 적용되어야 함을 의미합니다.

반환값

부울 값을 반환합니다. 다음 조건이 모두 참이면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

- 원격 사이트가 정의되어 있습니다.
- 문서 윈도우에 포커스가 있는 경우 파일이 로컬 사이트의 일부이고 아직 체크 아웃되지 않았으며, **[사이트]** 패널에 포커스가 있는 경우 하나 이상의 파일이나 폴더가 선택되어 있고 선택된 파일 중 하나 이상이 체크 아웃되지 않았습니까.
- 사이트에 대해 체크 인/체크 아웃 기능이 설정되어 있습니다.

site.canCloak()**지원 버전**

Dreamweaver MX

설명

Dreamweaver에서 [숨김] 작업을 수행할 수 있는지 여부를 확인합니다.

인수**siteOrURL**

- **siteOrURL** 인수는 **site** 키워드여야 합니다. 이는 **canCloak()** 함수가 **[사이트]** 패널의 선택 영역 또는 특정 폴더의 **URL**에 적용되어야 함을 나타냅니다. 즉, **canCloak()** 함수가 지정한 폴더 및 그 안의 모든 내용에 적용되어야 함을 의미합니다.

반환값

부울 값을 반환합니다. Dreamweaver에서 현재 사이트 또는 지정한 폴더에서 [숨김] 작업을 수행할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.canCompareFiles()**지원 버전**

Dreamweaver 8

설명

이 함수는 Dreamweaver에서 선택한 파일에 대해 **Compare** 함수를 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 두 파일(로컬 파일 하나와 원격 파일 하나, 로컬 파일 두 개 또는 원격 파일 두 개)이 선택되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.canConnect()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 원격 사이트에 연결할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 원격 사이트가 FTP 사이트이면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.canDisplaySyncInfoForFile()

지원 버전

Dreamweaver CS3

설명

Dreamweaver에서 `displaySyncInfoForFile` 작업을 수행할 수 있는지 여부를 확인합니다.

인수

`path`, `'site'`

- `path`는 로컬 파일에 대한 URL입니다.
- `'site'`는 해당 함수가 [사이트] 패널에 선택된 파일을 사용함을 나타냅니다.

반환값

로컬 파일 뷰에 선택된 파일이 있거나(`'site'`가 매개 변수인 경우), 전달된 경로가 사이트의 일부이면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.canGet()

지원 버전

Dreamweaver 3

설명

Dreamweaver에서 [내려받기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

`siteOrURL`

- `siteOrURL` 인수는 `site` 키워드여야 합니다. 이는 함수가 단일 파일의 URL 또는 [사이트] 패널의 선택 영역에 적용되어야 함을 의미합니다.

반환값

인수가 **site**인 경우에는 [사이트] 패널에서 하나 이상의 파일이나 폴더가 선택되어 있고 원격 사이트가 정의되어 있는지 여부를 나타내는 부울 값을 반환하고, 인수가 **URL**인 경우에는 원격 사이트가 정의된 사이트에 문서가 속해 있는지 여부를 나타내는 부울 값을 반환합니다.

site.canLocateInSite()**지원 버전**

Dreamweaver 3, CS4에서 업데이트됨

설명

Dreamweaver에서 인수에 따라 [로컬 사이트에서 찾기] 또는 [원격 사이트에서 찾기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

localOrRemote, siteOrURL

- **localOrRemote** 인수는 **local** 또는 **remote**여야 합니다.
- **siteOrURL** 인수는 **site** 키워드여야 합니다. 이 키워드는 함수가 단일 파일의 URL 또는 [사이트] 패널의 선택 영역에 적용되어야 함을 의미합니다.

반환값

다음 값 중 하나를 반환합니다.

- 문서가 사이트에 속하는지 여부를 나타내는 부울 값을 반환합니다. 첫 번째 인수가 **local** 키워드이고 두 번째 인수가 **URL**인 경우 부울 값이 반환됩니다.
- 부울 값을 반환합니다. 첫 번째 인수가 **remote** 키워드이고 두 번째 인수가 **URL**인 경우 부울 값이 반환됩니다. 부울 값은 다음을 나타냅니다.
 - 원격 사이트가 정의되어 있는 사이트에 문서가 속하는지 여부.
 - 서버 유형이 로컬/네트워크인 경우 하드 드라이브가 마운트되었는지 여부
- 두 윈도우 모두에 사이트 파일이 포함되어 있는지 여부와 선택 영역이 인수와 반대 구역에 있는지 여부를 나타내는 부울 값 두 번째 인수가 **site** 키워드인 경우 부울 값이 반환됩니다.

site.canMakeEditable()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [읽기 전용 해제] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. Dreamweaver에서 [읽기 전용 해제] 작업을 수행할 수 있으면 **true**를 반환하고 선택된 파일 중 하나 이상이 잠겨 있으면 **false**를 반환합니다.

site.canMakeNewFileOrFolder()**지원 버전**

Dreamweaver 3

설명

Dreamweaver가 [사이트] 패널에서 [새 파일] 또는 [새 폴더] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [사이트] 패널의 선택된 구역에 파일이 표시되면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.canOpen()**지원 버전**

Dreamweaver 3

설명

Dreamweaver가 [사이트] 패널에 현재 선택된 파일이나 폴더를 열 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. [사이트] 패널에 선택된 파일이나 폴더가 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.canPut()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [올리기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

siteOrURL

- **siteOrURL** 인수는 **site** 키워드여야 합니다. 이는 함수가 단일 파일의 URL 또는 [사이트] 패널의 선택 영역에 적용되어야 함을 의미합니다.

반환값

다음 값 중 하나를 반환합니다.

- 인수가 **site** 키워드인 경우, [사이트] 패널에 선택된 파일이나 폴더가 있고 원격 사이트가 정의되어 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.
- 인수가 URL인 경우, 원격 사이트가 정의된 사이트에 문서가 속해 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.canRecreateCache()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [사이트 캐시 재생성] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 현재 사이트에 대해 [링크 업데이트 속도 향상을 위해 캐시 사용] 옵션이 활성화되어 있으면 **true**를 반환합니다.

site.canRefresh()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [[로컬 | 원격] 새로 고침] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

localOrRemote

- **localOrRemote** 인수는 **local** 또는 **remote** 키워드여야 합니다.

반환값

localOrRemote 인수가 **local** 키워드이면 **true**를 반환하고 그렇지 않으면 원격 사이트가 정의되었는지 여부를 나타내는 부울 값을 반환합니다.

site.canSelectAllCheckedOutFiles()**지원 버전**

Dreamweaver 4

설명

현재 작업 중인 사이트에 체크 인/체크 아웃 기능이 활성화되어 있는지 여부를 확인합니다.

인수
없음

반환값

부울 값을 반환합니다. 사이트에 체크 인/체크 아웃을 사용할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.canSelectNewer()

지원 버전
Dreamweaver 3

설명

Dreamweaver에서 [최신 [로컬 | 원격] 선택] 작업을 수행할 수 있는지 여부를 확인합니다.

인수
localOrRemote

- **localOrRemote** 인수는 **local** 또는 **remote** 키워드여야 합니다.

반환값

원격 사이트가 정의되어 있는 사이트에 문서가 속해 있는지 여부를 나타내는 부울 값을 반환합니다.

site.canSynchronize()

지원 버전
Dreamweaver 3

설명

Dreamweaver에서 [동기화] 작업을 수행할 수 있는지 여부를 확인합니다.

인수
없음

반환값

원격 사이트가 정의되어 있는지 여부를 나타내는 부울 값을 반환합니다.

site.canUncloak()

지원 버전
Dreamweaver MX

설명

Dreamweaver에서 [숨김 해제] 작업을 수행할 수 있는지 여부를 확인합니다.

인수**siteOrURL**

- **siteOrURL** 인수는 **site** 키워드여야 합니다. 이는 **canUncloak()** 함수가 [사이트] 패널의 선택 영역 또는 특정 폴더의 URL에 적용되어야 함을 나타냅니다. 즉, **canUncloak()** 함수가 지정한 폴더 및 그 안의 모든 내용에 적용되어야 함을 의미합니다.

반환값

부울 값을 반환합니다. Dreamweaver에서 현재 사이트 또는 지정한 폴더에서 [숨김 해제] 작업을 수행할 수 있으면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

site.canUndoCheckOut()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [체크 아웃 실행 취소] 작업을 수행할 수 있는지 여부를 확인합니다.

인수**siteOrURL**

- **siteOrURL** 인수는 **site** 키워드여야 합니다. 이는 함수가 단일 파일의 URL 또는 [사이트] 패널의 선택 영역에 적용되어야 함을 의미합니다.

반환값

부울 값을 반환합니다. 지정된 파일이나 하나 이상의 선택된 파일이 체크 아웃되었으면 **true**를 반환합니다.

site.canViewAsRoot()**지원 버전**

Dreamweaver 3

설명

Dreamweaver에서 [루트로 보기] 작업을 수행할 수 있는지 여부를 확인합니다.

인수

없음

반환값

부울 값을 반환합니다. 지정된 파일이 HTML 또는 Flash 파일이면 **true**를 반환하고 그렇지 않으면 **false**를 반환합니다.

색인

A

activateApp() 109
 activeViewScale 363
 addBehavior() 275
 addDebugContextData() 146
 addItem() 141
 addJavaScript() 298
 addLinkToExistingFile() 203
 addResultItem() 137
 alert() 110
 align() 349
 API, 유형
 Fireworks 통합 33
 Flash 객체 39
 HTTP 16
 데이터베이스 53
 데이터베이스 연결 대화 상자 77
 디자인 노트 22
 소스 제어 통합 84
 파일 I/O 7
 applyCharacterMarkup() 391
 applyComment() 193
 applyConnection() 79
 applyCSSStyle() 331
 applyFontMarkup() 391
 applyLayout() 329
 applySelectedStyle() 337
 applyTemplate() 287
 arrange() 350
 arrangeFloatingPalettes() 357
 arrowDown() 127, 412
 arrowLeft() 128, 413
 arrowRight() 128, 413
 arrowUp() 129, 414
 assetPalette.addToFavoritesFromDocument() 268
 assetPalette.addToFavoritesFromSiteAssets() 268
 assetPalette.addToFavoritesFromSiteWindow() 269

assetPalette.canEdit() 441
 assetPalette.canInsertOrApply() 442
 assetPalette.canUpdateSmartObjectFromOriginal() 49
 assetPalette.copyToSite() 269
 assetPalette.edit() 269
 assetPalette.getSelectedCategory() 270
 assetPalette.getSelectedItems() 270
 assetPalette.getSelectedView() 271
 assetPalette.insertOrApply() 271
 assetPalette.locateInSite() 272
 assetPalette.newAsset() 272
 assetPalette.newFolder() 272
 assetPalette.recreateLibraryFromDocument() 273
 assetPalette.refreshSiteAssets() 273
 assetPalette.removeFromFavorites() 273
 assetPalette.renameNickname() 274
 assetPalette.setSelectedCategory() 274
 assetPalette.setSelectedView() 274
 assetPalette.updateSmartObjectFromOriginal() 49
 attachExternalStylesheet() 338

B

backspaceKey() 129
 balanceBracesTextView() 414
 beep() 110, 112, 198, 199, 226
 bridge 통신 함수 115
 BridgeTalk
 bringToFront() 115
 send() 115
 suppressStartupScreen() 116
 bringAttentionToFloater() 172
 bringDWToFront() 33
 bringFWToFront() 33
 bringToFront() 115
 browseDocument() 102, 459
 browseForFileURL() 233
 browseForFolderURL() 234

browseInBridge() 116
 browser.getSelection() 319
 browser.getStatusText() 319
 browser.getWindow() 320
 browserEle.loadHTML() 320

C

canAlign() 431
 canApplyLayout() 330
 canApplyTemplate() 431
 canArrange() 432
 canCheckIn() 459
 canCheckOut() 459
 canClear() 453
 canClipCopy() 442
 canClipCopyText() 432
 canClipCut() 442
 canClipPaste() 432, 443
 canClipPasteText() 433
 canCloak() 460
 canCompare() 456
 canCompareFiles() 460
 canConnect() 461
 canConvertLayersToTable() 433
 canConvertTablesToLayers() 433
 canCopy() 454
 canCut() 454
 canDecreaseColspan() 434
 canDecreaseRowspan() 434
 canDeleteSelection() 443
 canDeleteTableColumn() 434
 canDeleteTableRow() 435
 canDisplaySyncInfoForFile() 461
 canEditColumns() 435
 canEditNoFramesContent() 435
 canEditSelectedRule() 449
 canEditSelection() 453
 canEditSnippet() 458
 canExportTemplateDataAsXML() 443

- canFindNext() 444
- canFitSelection() 444
- canGet() 461
- canIncreaseColspan() 435
- canIncreaseRowspan() 436
- canInsert() 458
- canInsertTableColumns() 436
- canInsertTableRows() 436
- canLocateInSite() 462
- canMakeEditable() 462
- canMakeNewEditableRegion() 437
- canMakeNewFileOrFolder() 463
- canMarkDelete() 456
- canMarkGet() 457
- canMarkIgnore() 457
- canMarkPut() 457
- canMarkSelectionAsEditable() 437
- canMarkSynced() 458
- canMergeTableCells() 437
- canOpen() 463
- canOpenInBrowser() 455
- canOpenInEditor() 455
- canOpenInFrame() 444
- canPaste() 454
- canPasteSpecial() 445
- canPlayPlugin() 438
- canPlayRecordedCommand() 445
- canPopupEditTagDialog() 445
- canPut() 463
- canRecreateCache() 464
- canRedo() 438, 446
- canRefresh() 464
- canRemoveEditableRegion() 438
- canRenameSelectedStyle() 452
- canRevertDocument() 446
- canSave() 455
- canSaveAll() 446
- canSaveDocument() 447
- canSaveDocumentAsTemplate() 447
- canSaveFrameset() 447
- canSaveFramesetAs() 448
- canSelectAll() 448, 456
- canSelectAllCheckedOutFiles() 464
- canSelectNewer() 465
- canSelectTable() 439
- canSetLinkHref() 439
- canShowFindDialog() 448
- canShowListPropertiesDialog() 439
- canSplitFrame() 440
- canSplitTableCell() 440
- canStopPlugin() 440
- canSynchronize() 465
- canUncloak() 465
- canUndo() 441, 449
- canUndoCheckOut() 466
- canUpdateSmartObjectFromOriginal() 52
- canViewAsRoot() 466
- canZoom() 449
- cascade() 173
- changeLink() 204
- changeLinkSitewide() 203
- checkIn() 204
- checkLinks() 204
- checkOut() 205
- checkSpelling() 245
- checkTargetBrowsers() 205, 245
- cleanupXHTML() 231
- clear() 137
- clearGuides() 366
- clearServerScriptsFolder() 16
- clearSteps() 121
- clearTemp() 17
- clipCopy() 283, 285
- clipCopyText() 284
- clipCut() 284, 285
- clipPaste() 284, 286
- cloak() 206
- closeDocument() 234
- CloseNotesFile() 27
- codeHints.addFunction() 384
- codeHints.addMenu() 382
- codeHints.resetMenu() 385
- codeHints.showCodeHints() 385
- ColdFusion Administrator 63
- ColdFusion Component Explorer 212, 215
- ColdFusion 데이터 소스 61
- ColdFusion 데이터 소스 이름 54
- collapseFullTag() 188
- collapseSelectedCodeFragment() 187
- collapseSelectedCodeFragmentInverse() 187, 190
- collapseSelectedCodeFragment() 190
- compare() 200
- compareFiles() 199
- Configuration 폴더 경로 253
- Configuration/Temp 폴더 17, 18, 19
- connection_includefile.edml 81
- convertActiveContent() 126
- convertLayersToTable() 229
- convertNextActiveContent() 127
- convertTablesToLayers() 229
- convertToXHTML() 232
- convertWidthsToPercent() 373
- convertWidthsToPixels() 373
- Copy() 138
- copy() 7
- copyAssets() 298
- copySteps() 121
- createDocument() 235
- createFolder() 8
- createHorizontalGuide() 366
- createResultsWindow() 140
- createVerticalGuide() 367
- createXHTMLDocument() 235
- createXMLDocument() 236
- CSS 스타일 함수 329
- CSS(Cascading Style Sheet)를 HTML 마크업으로, 변환 229
- cssRuleTracker.canEditSelectedRule() 449
- cssStyle.canEditSelectedStyle() 451
- cssStylePalette.canApplySelectedStyle() 450
- cssStylePalette.canDeleteSelectedStyle() 450
- cssStylePalette.canDuplicateSelectedStyle() 451
- cssStylePalette.canEditStyleSheet() 452
- CSS를 HTML 마크업으로, 변환 229
- cut() 138

D

- dbi.setExpanded() 309
- decreaseColspan() 373
- decreaseRowspan() 374
- defineSites() 207
- Delete 키 129
- deleteConnection() 54
- deleteHorizontalGuide() 367
- deleteKey() 129
- deleteSelectedStyle() 338
- deleteSelection() 207, 392, 404
- deleteTableColumn() 374
- deleteTableRow() 374
- deleteVerticalGuide() 368
- deployFilesToTestingServerBin() 207
- detachFromLibrary() 287
- detachFromTemplate() 288
- displaySyncInfoForFile() 208
- div
 - ID 359, 361
 - 색상 표시 358, 360
 - 테두리 359, 361
- doDeferredTableUpdate() 375
- dom
 - addBehavior() 275
 - addJavaScript() 298
 - align() 349
 - applyCharacterMarkup() 391
 - applyCSSStyle() 331
 - applyFontMarkup() 391
 - applyLayout() 329
 - applyTemplate() 287
 - arrange() 350
 - arrowDown() 127
 - arrowLeft() 128
 - arrowRight() 128
 - arrowUp() 129
 - backspaceKey() 129
 - browser.getSelection() 319
 - browser.getWindow() 320
 - browser.interactivityPaused 320
 - browser.javaScriptEnabled 321
 - canAlign() 431
 - canApplyLayout() 330
 - canApplyTemplate() 431
 - canArrange() 432
 - canClipCopyText() 432
 - canClipPaste() 432
 - canClipPasteText() 433
 - canConvertLayersToTable() 433
 - canConvertTablesToLayers() 433
 - canDecreaseColspan() 434
 - canDecreaseRowspan() 434
 - canDeleteTableColumn() 434
 - canDeleteTableRow() 435
 - canEditNoFramesContent() 435
 - canIncreaseColspan() 435
 - canIncreaseRowspan() 436
 - canInsertTableColumns() 436
 - canInsertTableRows() 436
 - canMakeNewEditableRegion() 437
 - canMarkSelectionAsEditable() 437
 - canMergeTableCells() 437
 - canPlayPlugin() 438
 - canRedo() 438
 - canRemoveEditableRegion() 438
 - canSelectTable() 439
 - canSetLinkHref() 439
 - canShowListPropertiesDialog() 439
 - canSplitFrame() 440
 - canSplitTableCell() 440
 - canStopPlugin() 440
 - canUndo() 441
 - checkSpelling() 245
 - checkTargetBrowsers() 245
 - cleanupXHTML() 231
 - clearGuides() 366
 - clipCopy() 283
 - clipCopyText() 284
 - clipCut() 284
 - clipPaste() 284
 - collapseSelectedCodeFragment() 187
 - collapseSelectedCodeFragmentInverse() 187
 - convertActiveContent() 126
 - convertLayersToTable() 229
 - convertNextActiveContent() 127
 - convertTablesToLayers() 229
 - convertToXHTML() 232
 - convertWidthsToPercent() 373
 - convertWidthsToPixels() 373
 - copyAssets() 298
 - createHorizontalGuide() 366
 - createVerticalGuide() 367
 - decreaseColspan() 373
 - decreaseRowspan() 374
 - deleteHorizontalGuide() 367
 - deleteKey() 129
 - deleteSelection() 392
 - deleteTableColumn() 374
 - deleteTableRow() 374
 - deleteVerticalGuide() 368
 - detachFromLibrary() 287
 - detachFromTemplate() 288
 - doDeferredTableUpdate() 375
 - dreamweaver.absoluteURLToDocRelative() 252
 - editAttribute() 392
 - endOfDocument() 130
 - endOfLine() 130
 - exitBlock() 393
 - expandAllCodeFragments() 188
 - expandSelectedCodeFragments() 188
 - formatRange() 410
 - formatSelection() 410
 - getAttachedTemplate() 288
 - getAutoValidationCount() 411
 - getBehavior() 276
 - getBlockElements() 302
 - getCharset() 393
 - getClickedHeaderColumn() 357
 - getDefaultAssetFolder() 299
 - getDesignViewMode() 316
 - getEditableRegionList() 288
 - getEditNoFramesContent() 147
 - getElementView() 332
 - getFocus() 171
 - getFontMarkup() 393
 - getFrameNames() 347
 - getHeaderElements() 304
 - getHideAllVisualAids() 148
 - getInlineElements() 303
 - getIsLibraryDocument() 289
 - getIsTemplateDocument() 289

- getIsXHTMLDocument() 233
- getLineFromOffset() 394
- getLinkHref() 394
- getLinkTarget() 394
- getListElements() 304
- getListTag() 395
- getLiveViewDefaultsToUsingServer() 318
- getLiveViewDependentsUsingServer() 318
- getLiveViewUsingServer() 317
- getMinDisplayWidth() 301
- getOpenPathName() 192
- getParseMode() 245
- getPreventLayerOverlaps() 148
- getRulerOrigin() 352
- getRulerUnits() 352
- getSelectedEditableRegion() 289
- getSelectedNode() 255
- getSelection() 255
- getSelectorsDefinedInStylesheet() 256
- getShowAutoIndent() 148
- getShowBlockBackgrounds() 358
- getShowBlockBorders() 359
- getShowBlockIDs() 359
- getShowBoxModel() 360
- getShowDivBackgrounds() 332
- getShowDivBoxModel() 333
- getShowDivOutlines() 333
- getShowFrameBorders() 149
- getShowGrid() 149
- getShowHeaderView() 149
- getShowHiddenCharacters() 192
- getShowImageMaps() 150
- getShowInvalidHTML() 150
- getShowInvisibleElements() 150
- getShowLayerBorders() 151
- getShowLayoutTableTabs() 358
- getShowLayoutView() 358
- getShowLineNumbers() 151
- getShowNoscript() 410
- getShowRulers() 151
- getShowSyntaxColoring() 152
- getShowTableBorders() 152
- getShowTableWidths() 375
- getShowToolbar() 152
- getShowToolbarIconLabels() 166
- getShowTracingImage() 153
- getShowWordWrap() 153
- getSnapToGrid() 153
- getTableExtent() 375
- getTagSelectorTag() 426
- getTextAlignment() 395
- getTextFormat() 395
- getToolBarIdArray() 166
- getToolBarItemValue() 166
- getToolBarLabel() 167
- getToolBarVisibility() 168
- getTracingImageOpacity() 352
- getView() 171
- getWindowTitle() 172
- guidesColor() 368
- guidesDistanceColor() 369
- guidesLocked 369
- guidesSnapToElements 370
- guidesVisible 370
- hasCharacterMarkup() 396
- hasGuides 370
- hasHorizontalGuide() 371
- hasTracingImage() 441
- hasVerticalGuide() 371
- hideInfoBar() 182
- hideInfoMessagePopup() 246
- increaseColspan() 376
- increaseRowspan() 376
- indent() 396
- insertFiles() 109
- insertHTML() 396
- insertLibraryItem() 290
- insertObject() 397
- insertTableColumns() 376
- insertTableRows() 377
- insertText() 397
- isDesignViewUpdated() 411
- isDocumentInFrame() 348
- isRelatedFileViewOpen() 185
- isSelectionValid() 412
- loadTracingImage() 353
- makeSizesEqual() 350
- markSelectionAsEditable() 290
- mergeTableCells() 377
- moveSelectionBy() 350
- newBlock() 398
- newEditableRegion() 291
- nextParagraph() 130
- nextWord() 131
- nodeToOffsets() 256
- notifyFlashObjectChanged() 398
- offsetsToNode() 257
- outdent() 399
- pageDown() 131
- pageUp() 132
- playAllPlugins() 353
- playPlugin() 353
- previousParagraph() 132
- previousWord() 132
- reapplyBehaviors() 276
- redo() 117
- removeAllTableHeights() 378
- removeAllTableWidths() 378
- removeBehavior() 277
- removeCharacterMarkup() 399
- removeColumnWidth() 378
- removeCSSStyle() 334
- removeEditableRegion() 291
- removeFontMarkup() 399
- resetAllElementViews() 334
- resizeSelection() 400
- resizeSelectionBy() 351
- runTranslator() 264
- runValidation() 246
- saveAllFrames() 348
- selectAll() 258
- selectChild() 407
- selectParent() 408
- selectTable() 379
- serverModel.getAppURLPrefix() 323
- serverModel.getDelimiters() 324
- serverModel.getDisplayName() 324
- serverModel.getFolderName() 325
- serverModel.getServerIncludeUrlPatterns()
() 325
- serverModel.getServerInfo() 326
- serverModel.getServerName() 327
- serverModel.getServerSupportsCharset()
() 327

- serverModel.getServerVersion() 327
- serverModel.testAppServer() 328
- setAttributeWithErrorChecking() 400
- setDesignViewMode() 317
- setEditNoFramesContent() 154
- setElementView() 335
- setHideAllVisualAids() 154
- setLayerTag() 351
- setLinkHref() 400
- setLinkTarget() 401
- setListBoxKind() 401
- setListTag() 402
- setLiveViewDependentsUsingServer() 318
- setLiveViewUsingServer() 317
- setPreventLayerOverlaps() 154
- setRulerOrigin() 354
- setRulerUnits() 354
- setSelectedNode() 258
- setSelection() 258
- setShowBlockBackgrounds() 360
- setShowBlockBorders() 361
- setShowBlockIDs() 361
- setShowBoxModel() 362
- setShowDivBackgrounds() 335
- setShowDivBoxModel() 336
- setShowDivOutlines() 336
- setShowFrameBorders() 155
- setShowGrid() 155
- setShowHeaderView() 155
- setShowHiddenCharacters() 192
- setShowImageMaps() 156
- setShowInvalidHTML() 156
- setShowInvisibleElements() 156
- setShowLayerBorders() 157
- setShowLayoutTableTabs() 362
- setShowLayoutView() 362
- setShowLineNumbers() 157
- setShowNoscript() 412
- setShowRulers() 157
- setShowSyntaxColoring() 158
- setShowTableBorders() 158
- setShowTableWidths() 379
- setShowToolBar() 158
- setShowToolBarIconLabels() 169
- setShowTracingImage() 159
- setShowWordWrap() 159
- setSnapToGrid() 159
- setTableCellTag() 379
- setTableColumns() 380
- setTableRows() 380
- setTextAlignment() 402
- setTextFieldKind() 403
- setTextFormat() 403
- setToolBarItemAttribute() 168
- setToolBarPosition() 169
- setToolBarVisibility() 170
- setTracingImageOpacity() 355
- setTracingImagePosition() 355
- setView() 172
- showFontColorDialog() 403
- showInfoBar() 181
- showInfoMessagePopup() 247
- showInsertTableRowsOrColumnsDialog() 380
- showListPropertiesDialog() 402
- showLiveViewParametersDialog() 319
- showPagePropertiesDialog() 248
- snapToGuides() 372
- snapTracingImageToSelection() 356
- source.applyComment() 193
- source.arrowDown() 412
- source.arrowLeft() 413
- source.arrowRight() 413
- source.arrowUp() 414
- source.balanceBracesTextView() 414
- source.doCodeNavItem() 414
- source.endOfDocument() 415
- source.endOfLine() 415
- source.endPage() 415
- source.getCurrentLines() 416
- source.getLineFromOffset() 416
- source.getSelection() 416
- source.getText() 417
- source.getValidationErrorsForOffset() 417
- source.indentTextView() 418
- source.insert() 418
- source.nextWord() 418
- source.outdentTextView() 419
- source.pageDown() 419
- source.pageUp() 420
- source.previousWord() 420
- source.refreshVariableCodeHints() 193
- source.removeComment() 194
- source.replaceRange() 420
- source.scrollEndFile() 421
- source.scrollLineDown() 421
- source.scrollLineUp() 422
- source.scrollPageDown() 422
- source.scrollPageUp() 422
- source.scrollToFile() 423
- source.selectParentTag() 423
- source.setCurrentLine() 423
- source.startOfDocument() 424
- source.startOfLine() 424
- source.topPage() 424
- source.wrapSelection() 425
- splitFrame() 349
- splitTableCell() 381
- startOfDocument() 133
- startOfLine() 133
- stopAllPlugins() 356
- stopPlugin() 356
- stripTag() 408
- synchronizeDocument() 425
- undo() 117
- updateCurrentPage() 291
- wrapTag() 408
- DOM, 가져오기 237
- dom.browser.getStatusText() 319
- dom.browser.javaScriptEnabled 321
- dom.collapseFullTag() 185
- dom.collapseFullTagInverse() 186
- doURLDecoding() 248
- doURLEncoding() 260
- Dreamweaver
 - 맨 앞으로 가져오기 33
 - 종료 111
- dreamweaver
 - activateApp() 109
 - activeViewScale 363
 - arrangeFloatingPalettes() 357
 - assetPalette.addToFavoritesFromDocument() 268

- assetPalette.addToFavoritesFromSiteAssets() 268
- assetPalette.addToFavoritesFromSiteWindow() 269
- assetPalette.canEdit() 441
- assetPalette.canInsertOrApply() 442
- assetPalette.canUpdateSmartObjectFromOriginal() 49
- assetPalette.copyToSite() 269
- assetPalette.edit() 269
- assetPalette.getSelectedCategory() 270
- assetPalette.getSelectedItems() 270
- assetPalette.getSelectedView() 271
- assetPalette.insertOrApply() 271
- assetPalette.locateInSite() 272
- assetPalette.newAsset() 272
- assetPalette.newFolder() 272
- assetPalette.recreateLibraryFromDocument() 273
- assetPalette.refreshSiteAssets() 273
- assetPalette.removeFromFavorites() 273
- assetPalette.renameNickname() 274
- assetPalette.setSelectedCategory() 274
- assetPalette.setSelectedView() 274
- assetPalette.updateSmartObjectFromOriginal() 49
- beep() 110, 112, 198, 199, 226
- behaviorInspector 객체 275
- behaviorInspector.getBehaviorAt() 279
- behaviorInspector.getBehaviorCount() 280
- behaviorInspector.getSelectedBehavior() 280
- behaviorInspector.moveBehaviorDown() 280
- behaviorInspector.moveBehaviorUp() 281
- behaviorInspector.setSelectedBehavior() 282
- bringAttentionToFloater() 172
- browseDocument() 102
- browseForFileURL() 233
- browseForFolderURL() 234
- canClipCopy() 442
- canClipCut() 442
- canClipPaste() 443
- canDeleteSelection() 443
- canExportTemplateDataAsXML() 443
- canFindNext() 444
- canFitSelection() 444
- canOpenInFrame() 444
- canPasteSpecial() 445
- canPlayRecordedCommand() 445
- canPopupEditTagDialog() 445
- canRedo() 446
- canRevertDocument() 446
- canSaveAll() 446
- canSaveDocument() 447
- canSaveDocumentAsTemplate() 447
- canSaveFrameset() 447
- canSaveFramesetAs() 448
- canSelectAll() 448
- canShowFindDialog() 448
- canUndo() 449
- canUpdateSmartObjectFromOriginal() 52
- canZoom() 449
- cascade() 173
- clipCopy() 285
- clipCut() 285
- clipPaste() 286
- closeDocument() 234
- codeHints.addFunction() 384
- codeHints.addMenu() 382
- codeHints.resetMenu() 385
- codeHints.showCodeHints() 385
- compareFiles() 199
- createDocument() 235
- createResultsWindow() 140
- createXHTMLDocument() 235
- createXMLDocument() 236
- cssRuleTracker.canEditSelectedRule() 449
- cssRuleTracker.editSelectedRule() 336
- cssRuleTracker.newRule() 337
- cssStyle.canEditSelectedStyle() 451
- cssStylePalette 객체 329
- cssStylePalette.applySelectedStyle() 337
- cssStylePalette.canApplySelectedStyle() 450
- cssStylePalette.canDeleteSelectedStyle() 450
- cssStylePalette.canDuplicateSelectedStyle() 451
- cssStylePalette.canEditStyleSheet() 452
- cssStylePalette.deleteSelectedStyle() 338
- cssStylePalette.duplicateSelectedStyle() 338
- cssStylePalette.editSelectedStyle() 339
- cssStylePalette.editSelectedStyleInCodeview() 339
- cssStylePalette.editStyleSheet() 340
- cssStylePalette.getDisplayStyles() 340
- cssStylePalette.getMediaType() 341
- cssStylePalette.getSelectedStyle() 341
- cssStylePalette.getStyles() 342
- cssStylePalette.newStyle() 343
- cssStylePalette.renameSelectedStyle() 343
- cssStylePalette.setDisplayStyles() 344
- cssStylePalette.setMediaType() 344
- dbi.getDataSources() 309
- deleteSelection() 404
- doURLDecoding() 248
- doURLEncoding() 260
- editCommandList() 230
- editFontList() 404
- editLockedRegions() 264
- exportTemplateDataAsXML() 237
- findNext() 386
- fitAll() 363
- fitSelection 364
- fitWidth() 364
- flash.controlExists 48
- flash.evalScript() 47
- flash.executeScript() 47
- flash.newControl() 42
- flash.requestStateChange() 43
- flash.setMenu() 45
- getActiveRelatedFilePath() 183
- getActiveWindow() 173
- getAppBarColor() 196
- getBehaviorElement() 277
- getBehaviorTag() 278
- getBlockVisBoxModelColors() 345
- getBlockVisOutlineProperties() 345
- getBrowserList() 103
- getClipboardText() 286
- getConfigurationPath() 253
- getDivBackgroundColors() 346
- getDocumentDOM() 237

- getDocumentList() 174
- getDocumentPath() 253
- getElementRef() 249
- getExtDataArray() 310
- getExtDataValue() 310
- getExtensionEditorList() 103
- getExternalTextEditor() 104
- getExtGroups() 311
- getExtParticipants() 311
- getFlashPath() 104
- getFloatersVisible() 174
- getFloaterVisibility() 174
- getFocus() 176
- getFontList() 404
- getFontStyles() 405
- getHideAllFloaters() 160
- getImageHeight() 51
- getImageWidth() 51
- getKeyState() 405
- getLiveDataInitTags() 312
- getLiveDataMode() 313
- getLiveDataParameters() 313
- getMenuNeedsUpdating() 134
- getNaturalSize() 406
- getNewDocumentDOM() 238
- getPanelColor() 195
- getParticipants() 322
- getPreferenceInt() 249
- getPreferenceString() 250
- getPrimaryBrowser() 105
- getPrimaryExtensionEditor() 105
- getPrimaryView() 176, 184
- getRecentFileList() 238
- getRedoText() 118
- getRelatedFiles() 182
- getSecondaryBrowser() 105
- getServerModels() 328
- getShowDialogsOnInsert() 111
- getShowStatusBar() 160
- getSiteRoot() 254
- getSmartObjectOriginalWidth() 50
- getSmartObjectState() 50
- getSnapDistance() 176
- getSplitViewOrientation() 183
- getSystemFontList() 406
- getTempFolderPath() 254
- getTokens() 261
- getTranslatorList() 264
- getUndoText() 118
- historyPalette 객체 117
- historyPalette.clearSteps() 121
- historyPalette.copySteps() 121
- historyPalette.getSelectedSteps() 121
- historyPalette.getStepCount() 122
- historyPalette.getStepsAsJavaScript() 122
- historyPalette.getUndoState() 123
- historyPalette.replaySteps() 123
- historyPalette.saveAsCommand() 124
- historyPalette.setSelectedSteps() 124
- historyPalette.setUndoState() 125
- htmlInspector.collapseFullTag() 188
- htmlInspector.collapseSelectedCodeFragment() 190
- htmlInspector.collapseSelectedCodeFragmentInverse() 190
- htmlInspector.expandAllCodeFragments() 191
- htmlInspector.expandSelectedCodeFragments() 191
- htmlInspector.getShowAutoIndent() 161
- htmlInspector.getShowHiddenCharacters() 194
- htmlInspector.getShowHighlightInvalidHTML() 161
- htmlInspector.getShowLineNumbers() 161
- htmlInspector.getShowSyntaxColoring() 162
- htmlInspector.getShowWordWrap() 162
- htmlInspector.setShowAutoIndent() 162
- htmlInspector.setShowHiddenCharacters() 195
- htmlInspector.setShowHighlightInvalidHTML() 163
- htmlInspector.setShowLineNumbers() 163
- htmlInspector.setShowSyntaxColoring() 163
- htmlInspector.setShowWordWrap() 164
- htmlStylePalette.canEditSelection() 453
- importXMLIntoTemplate() 239
- isRecording() 453
- isReporting() 197
- latin1ToNative() 261
- libraryPalette 객체 287
- liveDataTranslate() 314
- loadSitesFromPrefs() 200
- mapKeyCodeToChar() 134
- minimizeRestoreAll() 177
- nativeToLatin1() 262
- newDocument() 239
- newFromTemplate() 240
- nodeExists() 259
- notifyMenuUpdated() 135
- objectPalette.getMenuDefault() 125
- objectPalette.setMenuDefault() 125
- openDocument() 240
- openDocumentFromSite() 240
- openInFrame() 241
- openRelatedFile() 183
- openWithApp() 107
- openWithBrowseDialog() 107
- openWithExternalTextEditor() 107
- openWithImageEditor() 108
- playRecordedCommand() 119
- popupAction() 279
- popupEditTagDialog() 427
- popupInsertTagDialog() 426
- popupServerBehavior() 323
- PrintCode() 407
- printDocument() 109
- quitApplication() 111
- redo() 119
- referencePalette.getFontSize() 275
- referencePalette.setFontSize() 275
- refreshExtData() 312
- relativeToAbsoluteURL() 254
- releaseDocument() 241
- reloadCodeColoring() 386
- reloadMenus() 135
- reloadObjects() 126
- reloadToolbars() 170
- replace() 387
- replaceAll() 387
- resolveOriginalAssetFileURLToAbsoluteLocalFilePath() 51
- resultsPalette.canClear() 453

- resultsPalette.canCopy() 454
- resultsPalette.canCut() 454
- resultsPalette.canOpenInBrowser() 455
- resultsPalette.canOpenInEditor() 455
- resultsPalette.canPaste() 454
- resultsPalette.canSave() 455
- resultsPalette.canSelectAll() 456
- resultsPalette.clear() 137
- resultsPalette.Copy() 138
- resultsPalette.cut() 138
- resultsPalette.debugWindow.addDebugContextData() 146
- resultsPalette.openInBrowser() 139
- resultsPalette.openInEditor() 139
- resultsPalette.paste() 138
- resultsPalette.save() 139
- resultsPalette.selectAll() 140
- resultsPalette.siteReports.addResultItem() 137
- revealDocument() 110
- revertDocument() 242
- runCommand() 230
- saveAll() 242
- saveDocument() 243
- saveDocumentAs() 243
- saveDocumentAsTemplate() 243
- saveFrameset() 244
- saveFramesetAs() 244
- saveSitesToPrefs() 200
- scanSourceString() 262
- selectAll() 260
- serverBehaviorInspector.getServerBehaviors() 322
- serverComponents.getSelectedNode() 308
- serverComponents.refresh() 308
- setActiveWindow() 177
- setBlockVisOutlineProperties() 346
- setDivBackgroundColors() 347
- setFloaterVisibility() 177
- setHideAllFloaters() 164
- setLiveDataError() 314
- setLiveDataMode() 315
- setLiveDataParameters() 315
- setPreferenceInt() 251
- setPreferenceString() 251
- setPrimaryView() 179, 184
- setShowStatusBar() 164
- setSnapDistance() 179
- setSplitViewOrientation() 184
- setUpComplexFind() 387
- setUpComplexFindReplace() 388
- setUpFind() 389
- setUpFindReplace() 389
- showAboutBox() 111
- showDynamicData() 112
- showFindDialog() 390
- showFindReplaceDialog() 391
- showGridSettingsDialog() 357
- showLiveDataDialog() 316
- showPreferencesDialog() 113
- showProperties() 179
- showQuickTagEditor() 409
- showReportsDialog() 197
- showResults() 136
- showTagChooser() 113, 427
- showTagLibraryEditor() 427
- showTargetBrowsersDialog() 252
- siteSyncDialog.canCompare() 456
- siteSyncDialog.canMarkDelete() 456
- siteSyncDialog.canMarkGet() 457
- siteSyncDialog.canMarkIgnore() 457
- siteSyncDialog.canMarkPut() 457
- siteSyncDialog.canMarkSynced() 458
- siteSyncDialog.compare() 200
- siteSyncDialog.markDelete() 201
- siteSyncDialog.markGet() 201
- siteSyncDialog.markIgnore() 201
- siteSyncDialog.markPut() 202
- siteSyncDialog.markSynced() 202
- siteSyncDialog.toggleShowAllFiles() 203
- snippetpalette.canEditSnippet() 458
- snippetpalette.canInsert() 458
- snippetPalette.editSnippet() 294
- snippetPalette.getCurrentSnippetPath() 293
- snippetPalette.insert() 294
- snippetPalette.insertSnippet() 295
- snippetPalette.newFolder() 293
- snippetPalette.newSnippet() 294
- snippetPalette.remove() 296
- snippetPalette.rename() 295
- startRecording() 119
- stopRecording() 120
- stylePalette.attachExternalStylesheet() 338
- tagLibrary.getImportedTagList() 429
- tagLibrary.getSelectedLibrary() 428
- tagLibrary.getSelectedTag() 428
- tagLibrary.getTagLibraryDOM() 428
- tagLibrary.importDTDOrSchema() 429
- templatePalette 객체 287
- tileHorizontally() 180
- tileVertically() 180
- toggleFloater() 180
- undo() 120
- updatePages() 292
- updateReference() 181
- updateSmartObjectFromOriginal() 52
- useTranslatedSource() 265
- validateFlash() 108
- zoomIn() 365
- zoomOut() 365
- Dreamweaver CS4에 새로 추가된 함수 2
- Dreamweaver 및 소스 제어
 - 통합 83
- Dreamweaver 종료 111
- dreamweaver.absoluteURLToDocRelative() 252
- dreamweaver.cssStylePalette
 - canRenameSelectedStyle() 452
- dreamweaver.cssStylePalette.canRenameSelectedStyle() 452
- dreamweaver.flash.controlEvent() 44
- dreamweaver.resultsPalette.bcc 136
- Dreamweaver를 맨 앞으로 가져오기 33
- DSN, ODBC 58, 59
- duplicateSelectedStyle() 338
- dw
 - browseInBridge() 116
 - dbi.setExpanded() 309
 - getFilesForLayout() 330
 - getLayoutDescriptions() 331
 - getLayoutNames() 331
 - registerIdleHandler() 113
 - revokeIdleHandler() 114
- DWfile DLL 7

DWfile.copy() 7
DWfile.createFolder() 8
DWfile.exists() 8
DWfile.getAttributes() 9
DWfile.getCreationDate() 10
DWfile.getCreationDateObj() 10
DWfile.getModificationDate() 9
DWfile.getModificationDateObj() 11
DWfile.getSize() 11
DWfile.listFolder() 11
DWfile.read() 12
DWfile.remove() 13
DWfile.setAttributes() 13
DWfile.write() 14

E

editAttribute() 392
editColumns() 208
editCommandList() 230
editFontList() 404
editLockedRegions() 264
editSelectedRule() 336
editSelectedStyle() 339
editSnippet() 294
editStyleSheet() 340
EDML 파일 함수 310
elem
 getBlockElements() 302
 getComputedStyleProp() 300
 getHeaderElements() 304
 getInlineElements() 303
 getListElements() 304
 isBlockElement() 305
 isHeaderElement() 306
 isInlineElement() 305
 isListElement() 307
element
 getTranslatedAttribute() 296
 getTranslatedClassName 297
 removeTranslatedAttribute() 296
 setTranslatedAttribute() 297
 translatedStyle 297
endOfDocument() 130, 415

endOffline() 130, 415
endPage() 415
execJsInFireworks() 34
exists() 8
exitBlock() 393
expandAllCodeFragments() 188, 191
expandSelectedCodeFragments() 188, 191
exportSite() 209
exportTemplateDataAsXML() 237
Extension Data Manager 310

F

FilePathToLocalURL() 27
findConnection() 78
findNext() 386
Fireworks
 JavaScript 실행 34
 JavaScript를 전달 34
 맨 앞으로 가져오기 33
 버전 36
 최적화 세션 35
Fireworks 통합 API
 bringDWToFront() 33
 bringFWToFront() 33
 execJsInFireworks() 34
 getJsResponse() 34
 mayLaunchFireworks() 35
 optimizeInFireworks() 35
 validateFireworks() 36
 정보 33
Fireworks를 맨 앞으로 가져오기 33
fitall() 363
fitSelection() 364
fitWidth() 364
Flash MX, 버전 확인 108
Flash 객체 API
 SWFFile.createFile() 39
 SWFFile.getNaturalSize() 40
 SWFFile.getObjectType() 40
 SWFFile.readFile() 41
 정보 39
Flash 객체 유형 40

Flash 객체 파일
 생성 39
 읽기 41
Flash 객체 파일 생성 39
Flash 내용, 원래 크기 40
Flash 패널 및 대화 상자 함수 41
Flash, 경로 104
flash.controlEvent() 44
flash.controlExists 48
flash.evalScript() 47
flash.executeScript() 47
flash.newControl() 42
flash.requestStateChange() 43
flash.setMenu() 45
font 태그 392
formatRange() 410
formatSelection() 410
FTP 로깅 135
FWLaunch.bringDWToFront() 33
FWLaunch.bringFWToFront() 33
FWLaunch.execJsInFireworks() 34
FWLaunch.getJsResponse() 34
FWLaunch.mayLaunchFireworks() 35
FWLaunch.optimizeInFireworks() 35
FWLaunch.validateFireworks() 36

G

get() 211
getActiveRelatedFilePath() 183
getActiveWindow() 173
getAppBarColor() 196
getAppServerAccessType() 211
getAppServerPathToFiles() 212
getAppURLPrefix() 323
getAppURLPrefixForSite() 212
getAttachedTemplate() 288
getAttributes() 9
getAutoValidationCount() 411
getBehavior() 276
getBehaviorAt() 279
getBehaviorCount() 280
getBehaviorElement() 277
getBehaviorTag() 278

getBlockElements() 302
getBlockVisBoxModelColors() 345
getBlockVisOutlineProperties() 345
getBrowserList() 103
getCharSet() 393
getCheckOutUser() 213
getCheckOutUserForFile() 213
getClickedHeaderColumn() 357
getClipboardText() 286
getCloakingEnabled() 214
getColdFusionDsnList() 54
getColumnAndTypeList() 65
getColumnList() 66
getColumns() 67
getColumnsOfTable() 67
getComputedStyleProp() 300
getConfigurationPath() 253
getConnection() 55
getConnectionList() 56
getConnectionName() 56
getConnectionState() 214
getConnectionString() 57
getCreationDate() 10
getCreationDateObj() 10
getCurrentLines() 416
getCurrentSite() 214
getDataSources() 309
getDeclaredStyle() 301
getDefaultAssetFolder() 299
getDelimiters() 324
getDesignViewMode() 316
getDisplayName() 324
getDivBackgroundColors() 346
getDocumentDOM() 237
getDocumentList() 174
getDocumentPath() 253
getDriverName() 57
getDynamicBindings() 53
getEditableRegionList() 288
getEditNoFramesContent() 147
getElementRef() 249
getExtDataArray() 310
getExtDataValue() 310
getExtensionEditorList() 103
getExternalTextEditor() 104
getExtGroups() 311
getExtParticipants() 311
getFile() 18
getFileCallback() 19
getFilesForLayout() 330
getFlashPath() 104
getFloatersVisible() 174
getFloaterVisibility() 174
getFocus() 171, 176, 215
getFolderName() 325
getFontList() 404
getFontMarkup() 393
getFontStyles() 405
getFrameNames() 347
getHeaderElements() 304
getHideAllFloaters() 160
getHideAllVisualAids() 148
getImageHeight() 51
getImageWidth() 51
getImportedTagList() 429
getInlineElements() 303
getIsLibraryDocument() 289
getIsTemplateDocument() 289
getIsXHTMLDocument() 233
getItem() 136, 142
getItemCount() 136, 142
getJsResponse() 34
getKeyState() 405
getLayoutDescriptions() 331
getLayoutNames() 331
getLineFromOffset() 394, 416
getLinkHref() 394
getLinkTarget() 394
getListElements() 304
getListTag() 395
getLiveDataInitTags() 312
getLiveDataMode() 313
getLiveDataParameters() 313
getLiveViewDependentsUsingServer() 31
8
getLiveViewUsingServer() 317
getLocalDsnList() 58
getLocalPathToFiles() 215
getLocalRootURL() 215
getMediaType() 341
getMenuDefault() 125
getMenuNeedsUpdating() 134
getMinDisplayWidth() 301
getModificationDate() 9
getModificationDateObj() 11
getNaturalSize() 406
getNewDocumentDOM() 238
GetNote() 28
GetNoteLength() 28
GetNotesKeyCount() 29
GetNotesKeys() 29
getOpenpathName() 192
getPanelColor() 195
getParseMode() 245
getParticipants() 322
getPassword() 58
getPreferenceInt() 249
getPreferenceString() 250
getPreventLayerOverlaps() 148
getPrimaryBrowser() 105
getPrimaryExtensionEditor() 105
getPrimaryKeys() 68
getPrimaryView() 176, 184
getProcedures() 68
getRdsPassword() 58
getRdsUserName() 59
getRecentFileList() 238
getRedoText() 118
getRelatedFiles() 182
getRemoteDsnList() 59
getRulerOrigin() 352
getRulerUnits() 352
getRuntimeConnectionType() 59
getSecondaryBrowser() 105
getSelectedBehavior() 280
getSelectedEditableRegion() 289
getSelectedItem() 136
getSelectedLibrary() 428
getSelectedNode() 255, 308

getSelectedSteps() 121
getSelectedStyle() 341
getSelectedTag() 428
getSelection() 216, 255, 416
getSelectorDefinedInStylesheet() 256
getServerBehaviors() 322
getServerIncludeUrlPatterns() 325
getServerInfo() 326
getServerModels() 328
getServerName() 327
getServerSupportsCharset() 327
getServerVersion() 327
getShowAutoIndent() 148
getShowBlockBackgrounds() 358
getShowBlockBorders() 359
getShowBlockIDs() 359
getShowBoxModel() 360
getShowDialogsOnInsert() 111
getShowFrameBorders() 149
getShowGrid() 149
getShowHeaderView() 149
getShowHiddenCharacters() 192, 194
getShowImageMaps() 150
getShowInvalidHTML() 150
getShowInvisibleElements() 150
getShowLayerBorders() 151
getShowLayoutTableTabs() 358
getShowLayoutView() 358
getShowLineNumbers() 151
getShowNoscript() 410
getShowRulers() 151
getShowStatusBar() 160
getShowSyntaxColoring() 152
getShowTableBorders() 152
getShowTableWidths() 375
getShowToolbar() 152
getShowToolbarIconLabels() 166
getShowToolTips() 165
getShowTracingImage() 153
getShowWordWrap() 153
getSiteForURL() 216
getSiteRoot() 254
GetSiteRootForFile() 30

getSiteRootForURL() 217
getSites() 216
getSiteURLPrefix() 217
getSize() 11
getSmartObjectOriginalWidth() 50
getSmartObjectState() 50
getSnapDistance() 176
getSnapToGrid() 153
getSPColumnList() 69
getSPColumnListNamedParams() 70
getSplitViewOrientation() 183
getSPParameters() 71
getSPParamsAsString() 72
getStepCount() 122
getStepsAsJavaScript() 122
getStyles() 342
getSystemFontList() 406
getTableExtent() 375
getTables() 72
getTagLibraryDOM() 428
getTagSelectorTag() 426
getTempFolderPath() 254
getText() 417
getTextAlignment() 395
getTextCallback() 20
getTextFormat() 395
getTokens() 261
getToolBarIdArray() 166
getToolBarItemValue() 166
getToolBarLabel() 167
getToolBarVisibility() 168
getTracingImageOpacity() 352
getTranslatedAttribute() 296
getTranslatedClassName 297
getTranslatorList() 264
getUndoState() 123
getUndoText() 118
getUserName() 60
getValidationErrorsForOffset() 417
GetVersionName() 30
GetVersionNum() 30
getView() 171
getViews() 73

getWindowTitle() 172
getXML() 265
getXMLSchema() 266
getXMLSourceURI() 266
guidesColor() 368
guidesDistanceColor() 369
guidesLocked 369
guidesSnapToElements 370
guidesVisible 370

H

hasCharacterMarkup() 396
hasConnectionWithName() 60
hasGuides() 370
hasHorizontalGuide() 371
hasTracingImage() 441
hasVerticalGuide() 371
hideInfoBar() 182
hideInfoMessagePopup() 246

HTML

CSS(Cascading Style Sheet) 229
XHTML로 변환 232
삼입 396
새 문서 만들기 235
연결 79
잘못된 표시 150
태그 351
htmlInspector.collapseFullTag() 188
htmlInspector.collapseSelectedCodeFragment() 190
htmlInspector.collapseSelectedCodeFragmentInverse() 190
htmlInspector.expandAllCodeFragments() 191
htmlInspector.expandSelectedCodeFragments() 191
htmlInspector.getShowAutoIndent() 161
htmlInspector.getShowHiddenCharacters() 194
htmlInspector.getShowHighlightInvalidHTML() 161
htmlInspector.getShowLineNumbers() 161
htmlInspector.getShowSyntaxColoring() 162

htmlInspector.getShowWordWrap() 162
 htmlInspector.setShowAutoIndent() 162
 htmlInspector.setShowHiddenCharacters()
 () 195
 htmlInspector.setShowHighlightInvalidH
 TML() 163
 htmlInspector.setShowLineNumbers() 16
 3
 htmlInspector.setShowSyntaxColoring()
 163
 htmlInspector.setShowWordWrap() 164
 HTTP API
 MMHttp.clearServerScriptsFolder() 16
 MMHttp.clearTemp() 17
 MMHttp.getFile() 18
 MMHttp.getFileCallback() 19
 MMHttp.getTextCallback() 20
 MMHttp.postText() 20
 MMHttp.postTextCallback() 21
 정보 16
 HTTP Post 20, 21
 httpReply 객체의 data 속성 16

I

ID 문자열, 제거 125
 importDTDOrSchema() 429
 importSite() 218
 importXMLIntoTemplate() 239
 increaseColspan() 376
 increaseRowspan() 376
 indent() 396
 indentTextView() 418
 InfoPrefs 구조 30
 insert() 294, 418
 insertFiles() 109
 insertHTML() 396
 insertLibraryItem() 290
 insertObject() 397
 insertSnippet() 295
 insertTableColumns() 376
 insertTableRows() 377
 insertText() 397
 inspectConnection() 79
 interactivityPaused 320

isBlockElement() 305
 isCloaked() 218
 isDesignViewUpdated() 411
 isDocumentInFrame() 348
 isHeaderElement() 306
 isInlineElement() 305
 isListElement() 307
 isRecording() 453
 isRelatedFileViewOpen() 185
 isReporting() 197
 isSelectionValid() 412
 itemInfo 구조체 86

J

JavaScript
 Fireworks에 전달 34
 Fireworks에서 실행 34
 JavaScript를 Fireworks에 전달 34
 JDBC 드라이버 57
 JDBC 연결 57

L

latin1ToNative() 261
 launchXMLSourceDialog() 267
 listFolder() 11
 liveDataTranslate() 314
 loadSitesFromPrefs() 200
 loadTracingImage() 353
 LocalURLToFilePath() 31
 locateInSite() 218

M

makeEditable() 219
 makeNewDreamweaverFile() 219
 makeNewFolder() 220
 makeSizesEqual() 350
 mapKeyCodeToChar() 134
 markDelete() 201
 markGet() 201
 markIgnore() 201
 markPut() 202
 markSelectionAsEditable() 290
 markSynced() 202

mayLaunchFireworks() 35
 menus.xml 파일 135
 mergeTableCells() 377
 minimizeRestoreAll() 177
 mm
 browsercontrol 321
 MMDB.deleteConnection() 54
 MMDB.getColdFusionDsnList() 54
 MMDB.getColumnAndTypeList() 65
 MMDB.getColumnList() 66
 MMDB.getColumns() 67
 MMDB.getColumnsOfTable() 67
 MMDB.getConnection() 55
 MMDB.getConnectionList() 56
 MMDB.getConnectionName() 56
 MMDB.getConnectionString() 57
 MMDB.getDriverName() 57
 MMDB.getLocalDsnList() 58
 MMDB.getPassword() 58
 MMDB.getPrimaryKeys() 68
 MMDB.getProcedures() 68
 MMDB.getRdsPassword() 58
 MMDB.getRdsUserName() 59
 MMDB.getRemoteDsnList() 59
 MMDB.getRuntimeConnectionType() 59
 MMDB.getSPColumnList() 69
 MMDB.getSPColumnListNamedParams(
) 70
 MMDB.getSPParameters() 71
 MMDB.getSPParamsAsString() 72
 MMDB.getTables() 72
 MMDB.getUserName() 60
 MMDB.getViews() 73
 MMDB.hasConnectionWithName() 60
 MMDB.needToPromptForRdsInfo() 61
 MMDB.needToRefreshColdFusionDsnLi
 st() 61
 MMDB.popupConnection() 61
 MMDB.setRdsPassword() 62
 MMDB.setRdsUserName() 62
 MMDB.showColdFusionAdmin() 63
 MMDB.showConnectionMgrDialog() 63
 MMDB.showOdbcDialog() 63
 MMDB.showRdsUserDialog() 64

MMDB.showRestrictDialog() 64
MMDB.showResultSet() 74
MMDB.showSPResultSet() 74
MMDB.showSPResultSetNamedParams()
75
MMDB.testConnection() 64
MMHttp.clearServerScriptsFolder() 16
MMHttp.clearTemp() 17
MMHttp.getFile() 18
MMHttp.getFileCallback() 19
MMHttp.getTextCallback() 20
MMHttp.postText() 20
MMHttp.postTextCallback() 21
MMNotes DLL 22
MMNotes 객체 22
MMNotes 공유 라이브러리
버전 번호 25, 30
버전 이름 25, 30
MMNotes.open() 26
MMNotes.remove() 26
MMNotes.set() 26
MMXSLT.getXML() 265
MMXSLT.getXMLSchema() 266
MMXSLT.getXMLSourceURI() 266
MMXSLT.launchXMLSourceDialog() 267
moveBehaviorDown() 280
moveBehaviorUp() 281
moveSelectionBy() 350

N

nativeToLatin1() 262
needToPromptForRdsInfo() 61
needToRefreshColdFusionDsnList() 61
newBlock() 398
newDocument() 239
newEditableRegion() 291
newFromTemplate() 240
newRule() 337
newSite() 220
newSnippet() 294
newStyle() 343
nextParagraph() 130
nextWord() 131, 418

nodeExists() 259
nodeToOffsets() 256
_notes 폴더 22
notifyFlashObjectChanged() 398
notifyMenuUpdated() 135

O

ODBC DSN 58, 59
ODBC 관리 63
offsetsToNode() 257
open() 26, 220
openDocument() 240
openDocumentFromSite() 240
openInBrowser() 139
openInEditor() 139
openInFrame() 241
OpenNotesFile() 31
OpenNotesFileWithOpenFlags() 31
openRelatedFile() 183
openWithApp() 107
openWithBrowseDialog() 107
openWithExternalTextEditor() 107
openWithImageEditor() 108
optimizeInFireworks() 35
outdent() 399
outdentTextView() 419

P

Page Down 131
Page Up 132
pageDown() 131, 419
pageUp() 132, 420
PanelWindow 사양 42
paste() 138
Photoshop 통합 49
playAllPlugins() 353
playPlugin() 353
playRecordedCommand() 119
popupAction() 279
popupConnection() 61
popupEditTagDialog() 427
popupInsertTagDialog() 426
popupServerBehavior() 323

postText() 20
postTextCallback() 21
previousParagraph() 132
previousWord() 132, 420
PrintCode() 407
printDocument() 109
put() 221

Q

quitApplication() 111

R

RDS

로그인 정보 61, 64
사용자 이름 59, 62
암호 58, 62

read() 12
reapplyBehaviors() 276
recreateCache() 221
redo() 117, 119
referencePalette.getFontSize() 275
referencePalette.setFontSize() 275
refresh() 222, 308
refreshExtData() 312
refreshVariableCodeHints() 193
registerIdleHandler() 113
relativeToAbsoluteURL() 254
releaseDocument() 241
reloadCodeColoring() 386
reloadMenus() 135
reloadObjects() 126
reloadToolbars() 170
remoteIsValid() 222
remove()
(dreamweaver.snippetPalette.remove
) 296
remove() (DWfile.remove) 13
remove() (MMNotes.remove) 26
removeAllTableHeights() 378
removeAllTableWidths() 378
removeBehavior() 277
removeCharacterMarkup() 399
removeColumnWidth() 378

removeComment() 194
removeCSSStyle() 334
removeEditableRegion() 291
removeFontMarkup() 399
RemoveNote() 32
removeTranslatedAttribute() 296
rename() 295
renameSelectedStyle() 343
renameSelection() 222
replace() 387
replaceAll() 387
replaceRange() 420
replaySteps() 123
resizeSelection() 400
resizeSelectionBy() 351
resolveOriginalAssetFileURLToAbsoluteLocalFilePath() 51
resultsPalette.canClear() 453
resultsPalette.canCopy() 454
resultsPalette.canCut() 454
resultsPalette.canOpenInBrowser() 455
resultsPalette.canOpenInEditor() 455
resultsPalette.canPaste() 454
resultsPalette.canSave() 455
resultsPalette.canSelectAll() 456
resultsPalette.clear() 137
resultsPalette.Copy() 138
resultsPalette.cut() 138
resultsPalette.debugWindow.addDebugContextData() 146
resultsPalette.openInBrowser() 139
resultsPalette.openInEditor() 139
resultsPalette.paste() 138
resultsPalette.save() 139
resultsPalette.selectAll() 140
resultsPalette.siteReports.addResultItem() 137
resWin.addItem() 141
resWin.addResultItem() 137
resWin.getItem() 142
resWin.getSelectedItem() 142
resWin.setButtons() 143
resWin.setCallbackCommands() 143
resWin.setColumnWidths() 143

resWin.setFileList() 144
resWin.setSelectedItem() 144
resWin.setTitle() 144
resWin.startProcessing() 145
resWin.stopProcessing() 145
revealDocument() 110
revertDocument() 242
revokeIdleHandler() 114
runCommand() 230
runTranslator() 264
runValidation() 223, 246

S

save() 139
saveAll() 242
saveAllFrames() 348
saveAsCommand() 124
saveDocument() 243
saveDocumentAs() 243
saveDocumentAsTemplate() 243
saveFrameset() 244
saveFramesetAs() 244
saveSitesToPrefs() 200
scanSourceString() 262
scrollEndFile() 421
scrollLineDown() 421
scrollLineUp() 422
scrollPageDown() 422
scrollPageUp() 422
scrollToFile() 423
SCS API. 소스 제어 통합 API 참조
SCS_AfterPut() 100, 101
SCS_BeforeGet() 99
SCS_BeforePut() 100
SCS_canCheckin() 98
SCS_canCheckout() 97
SCS_canConnect() 96
SCS_canDelete() 99
SCS_canGet() 97
SCS_canNewFolder() 98
SCS_canPut() 97
SCS_canRename() 99
SCS_CanUndoCheckout() 98

SCS_Checkin() 91
SCS_Checkout() 92
SCS_Connect() 84
SCS_Delete() 88
SCS_Disconnect() 84
SCS_Get() 87
SCS_GetAgentInfo() 84
SCS_GetCheckoutName() 91
SCS_GetConnectionInfo() 89
SCS_GetDesignNotes() 95
SCS_GetErrorMessage() 94
SCS_GetErrorMessageLength() 93
SCS_GetFileCheckoutList() 93
SCS_GetFolderList() 86
SCS_GetFolderListLength() 86
SCS_GetMaxNoteLength() 94
SCS_GetNewFeatures() 90
SCS_GetNoteCount() 94
SCS_GetNumCheckedOut() 93
SCS_GetNumNewFeatures() 90
SCS_GetRootFolder() 85
SCS_GetRootFolderLength() 85
SCS_IsConnected() 85
SCS_IsRemoteNewer() 96
SCS_ItemExists() 88
SCS_NewFolder() 87
SCS_Put() 87
SCS_Rename() 88
SCS_SetDesignNotes() 95
SCS_SiteDeleted() 89
SCS_SiteRenamed() 90
SCS_UndoCheckout() 92
SELECT 65, 66
selectAll() 140, 223, 258, 260
selectChild() 407
selectNewer() 223, 224
selectParent() 408
selectParentTag() 423
selectTable() 379
send() 115
serverdebuginfo 태그 146
set() 26
setActiveWindow() 177

- setAttributes() 13
- setAttributeWithErrorChecking() 400
- setBlockVisOutlineProperties() 346
- setCallbackCommands() 143
- setCloakingEnabled() 224
- setColumnWidths() 143
- setConnectionState() 225
- setCurrentLine() 423
- setCurrentSite() 225
- setDesignViewMode() 317
- setDivBackgroundColors() 347
- setEditNoFramesContent() 154
- setExpanded() 309
- setFileList() 144
- setFloaterVisibility() 177
- setFocus() 225
- setHideAllFloaters() 164
- setHideAllVisualAids() 154
- setLayerTag() 351
- setLinkHref() 400
- setLinkTarget() 401
- setListBoxKind() 401
- setListTag() 402
- setLiveDataError() 314
- setLiveDataMode() 315
- setLiveDataParameters() 315
- setLiveViewDependentsUsingServer() 318
- setLiveViewUsingServer() 317
- setMediaType() 344
- setMenuDefault() 125
- SetNote() 32
- setPreferenceInt() 251
- setPreferenceString() 251
- setPreventLayerOverlaps() 154
- setPrimaryView() 179, 184
- setRdsPassword() 62
- setRdsUserName() 62
- setRulerOrigin() 354
- setRulerUnits() 354
- setSelectedBehavior() 282
- setSelectedItem() 136
- setSelectedNode() 258
- setSelectedSteps() 124
- setSelection() 226, 258
- setShowBlockBackgrounds() 360
- setShowBlockBorders() 361
- setShowBlockIDs() 361
- setShowBoxModel() 362
- setShowFrameBorders() 155
- setShowGrid() 155
- setShowHeaderView() 155
- setShowHiddenCharacters() 192, 195
- setShowImageMaps() 156
- setShowInvalidHTML() 156
- setShowInvisibleElements() 156
- setShowLayerBorders() 157
- setShowLayoutTableTabs() 362
- setShowLayoutView() 362
- setShowLineNumbers() 157
- setShowNoscript() 412
- setShowRulers() 157
- setShowStatusBar() 164
- setShowSyntaxColoring() 158
- setShowTableBorders() 158
- setShowTableWidths() 379
- setShowToolBar() 158
- setShowToolBarIconLabels() 169
- setShowToolTips() 165
- setShowTracingImage() 159
- setShowWordWrap() 159
- setSnapDistance() 179
- setSnapToGrid() 159
- setSplitViewOrientation() 184
- setTableCellTag() 379
- setTableColumns() 380
- setTableRows() 380
- setTextAlignment() 402
- setTextFieldKind() 403
- setTextFormat() 403
- setTitle() 144
- setToolBarItemAttribute() 168
- setToolBarPosition() 169
- setToolBarVisibility() 170
- setTracingImageOpacity() 355
- setTracingImagePosition() 355
- setTranslatedAttribute() 297
- setUndoState() 125
- setUpComplexFind() 387
- setUpComplexFindReplace() 388
- setUpFind() 389
- setUpFindReplace() 389
- setView() 172
- showAboutBox() 111
- showColdFusionAdmin() 63
- showConnectionMgrDialog() 63
- showDynamicData() 112
- showFindDialog() 390
- showFindReplaceDialog() 391
- showFontColorDialog() 403
- showGridSettingsDialog() 357
- showInfoBar() 181
- showInfoMessagePopup() 247
- showInsertTableRowsOrColumnsDialog()
() 380
- showListPropertiesDialog() 402
- showLiveDataDialog() 316
- showLiveViewParamatersDialog() 319
- showOdbcDialog() 63
- showPagePropertiesDialog() 248
- showPreferencesDialog() 113
- showProperties() 179
- showQuickTagEditor() 409
- showRdsUserDialog() 64
- showReportsDialog() 197
- showRestrictDialog() 64
- showResults() 136
- showResultSet() 74
- showSPResultSet() 74
- showSPResultSetNamedParams() 75
- showTagChooser() 113, 427
- showTagLibraryEditor() 427
- showTargetBrowsersDialog() 252
- site
 - addLinkToExistingFile() 203
 - browseDocument() 459
 - canCheckIn() 459
 - canCheckOut() 459
 - canCloak() 460
 - canCompareFiles() 460

- canConnect() 461
- canDisplaySyncInfoForFile() 461
- canEditColumns() 435
- canGet() 461
- canLocateInSite() 462
- canMakeEditable() 462
- canMakeNewFileOrFolder() 463
- canOpen() 463
- canPut() 463
- canRecreateCache() 464
- canRefresh() 464
- canSelectAllCheckedOutFiles() 464
- canSelectNewer() 465
- canSynchronize() 465
- canUncloak() 465
- canUndoCheckOut() 466
- canViewAsRoot() 466
- changeLink() 204
- changeLinkSitewide() 203
- checkIn() 204
- checkLinks() 204
- checkOut() 205
- checkTargetBrowsers() 205
- cloak() 206
- defineSites() 207
- deleteSelection() 207
- deployFilesToTestingServerBin() 207
- displaySyncInfoForFile() 208
- editColumns() 208
- exportSite() 209
- get() 211
- getAppServerAccessType() 211
- getAppServerPathToFiles() 212
- getAppURLPrefixForSite() 212
- getCheckOutUser() 213
- getCheckOutUserForFile() 213
- getCloakingEnabled() 214
- getConnectionState() 214
- getCurrentSite() 214
- getFocus() 215
- getLocalPathToFiles() 215
- getLocalRootURL() 215
- getSelection() 216
- getShowToolTips() 165
- getSiteForURL() 216
- getSiteRootForURL() 217
- getSites() 216
- importSite() 218
- isCloaked() 218
- locateInSite() 218
- makeEditable() 219
- makeNewDreamweaverFile() 219
- makeNewFolder() 220
- newSite() 220
- open() 220
- put() 221
- recreateCache() 221
- refresh() 222
- remoteIsValid() 222
- renameSelection() 222
- runValidation() 223
- selectAll() 223
- selectNewer() 223, 224
- setCloakingEnabled() 224
- setConnectionState() 225
- setCurrentSite() 225
- setFocus() 225
- setSelection() 226
- setShowToolTips() 165
- synchronize() 227
- uncloak() 227
- uncloakAll() 227
- undoCheckOut() 228
- site.getSiteURLPrefix() 217
- siteSyncDialog.canCompare() 456
- siteSyncDialog.canMarkDelete() 456
- siteSyncDialog.canMarkGet() 457
- siteSyncDialog.canMarkIgnore() 457
- siteSyncDialog.canMarkPut() 457
- siteSyncDialog.canMarkSynced() 458
- siteSyncDialog.compare() 200
- siteSyncDialog.markDelete() 201
- siteSyncDialog.markGet() 201
- siteSyncDialog.markIgnore() 201
- siteSyncDialog.markPut() 202
- siteSyncDialog.markSynced() 202
- siteSyncDialog.toggleShowAllFiles() 203
- snapToGuides() 372
- snapTracingImageToSelection() 356
- snippet 태그, 속성 293
- snippetpalette.canEditSnippet() 458
- snippetpalette.canInsert() 458
- snippetPalette.getCurrentSnippetPath() 293
- snippetPalette.newFolder() 293
- source.applyComment() 193
- source.doCodeNavItem() 414
- source.refreshVariableCodeHints() 193
- source.removeComment() 194
- splitFrame() 349
- splitTableCell() 381
- Spry
 - 위젯 삽입 함수 298
 - 위젯 편집 함수 296
- SQL SELECT 65, 66
- SQL 문 74
 - 결과 표시 74
 - 열 가져오기 65, 66
- startOfDocument() 133, 424
- startOfLine() 133, 424
- startProcessing() 145
- startRecording() 119
- statusCode 속성 16
- stopAllPlugins() 356
- stopPlugin() 356
- stopProcessing() 145
- stopRecording() 120
- stripTag() 408
- suppressStartupScreen() 116
- SWFFile.createFile() 39
- SWFFile.getNaturalSize() 40
- SWFFile.getObjectType() 40
- SWFFile.readFile() 41
- synchronize() 227
- synchronizeDocument() 425

T

- testAppServer() 328
- testConnection() 64
- tileHorizontally() 180
- tileVertically() 180
- toggleFloater() 180

toggleShowAllFiles() 203

topPage() 424

translatedStyle 297

U

uncloak() 227

uncloakAll() 227

undo() 117, 120

undoCheckOut() 228

updateCurrentPage() 291

updatePages() 292

updateReference() 181

updateSmartObjectFromOriginal() 52

URL

Flash MX 응용 프로그램 104

데이터 게시 20

디코딩 248

브라우저에서 열기 102

상대 255

절대 파일 URL 255

파일 가져오기 18, 19

파일 내용 가져오기 19

userxd5 구성 파일 7

useTranslatedSource() 265

V

validateFireworks() 36

validateFlash() 108

W

window.getDeclaredStyle() 301

wrapSelection() 425

wrapTag() 408

write() 14

X

XHTML

만들기 235

문서 테스트 233

변환 232

정리 231

XHTML 문서, 정리 231

XHTML로 변환 232

XML 파일

가져오기 239

만들기 236

코드 단편 292

Z

zoomIn() 365

zoomOut() 365

ㄱ

가로 셀, 나누기 373

가져오기

해당 이름의 연결 객체 55

현재 DOM 237

객체 삽입 함수 125

객체 삽입시 대화 상자 보기 111

객체 유형, Flash 40

검색 135

게시

데이터 20

텍스트 21

결과 부동 패널 136

결과 윈도우

() 호출 143

결과 항목 추가 137

만들기 140

버튼 설정 143

선택한 항목 설정 144

선택한 항목의 인덱스 가져오기 142

열 폭 143

제목 144

추가 141

파일 처리 144, 145

함수 135

항목 배열 가져오기 142

항목 수 가져오기 142

결과 집합 70, 74, 75

결과 패널

메시지 138

지우기 137

결과 패널 그룹 135

경로

Configuration 폴더 253

Flash MX 응용 프로그램 104

문서 253

보조 브라우저 105

임시 폴더 254

경로 함수 252

고유 식별자 68

관련 파일 함수 182

구문 색상 표시 163

구성 파일 7

규칙, 설명서 5

기능, 소스 제어 시스템 90

기록

단계 119

중단 120

기록된 명령 119

기본 브라우저 105

보고서 139

기본 키 68

L

나누기

가로 셀 373

세로 셀 374

눈금자

단위 352

원점 352

ㄷ

다시 실행 119

다음 단락 131

다음 단어 131

단계

작업 내역 패널의 122

저장 124

단계 다시 실행 117

닫기

디자인 노트 파일 22, 27

문서 234

대화 상자

ColdFusion Administrator 63

ODBC 데이터 원본 관리자 63

격자 설정 357

다른 이름으로 저장 242, 243

대상 브라우저 252

동적 데이터 112

- 동적 텍스트 112
- 레이어를 표로 변환 229
- 명령 목록 편집 230
- 명령으로 저장 124
- 바꾸기 391
- 새 CSS 스타일 337
- 새 문서 239
- 새 스타일 343
- 선택하여 붙여넣기 112
- 스타일 시트 편집 340
- 스타일 정의 339
- 시스템 ODBC 관리 63
- 연결 관리자 63
- 외부 편집기 선택 107
- 이미지 소스 선택 353
- 정보 111
- 제한 64
- 찾기 390
- 크기 정의 146
- 태그 선택기 113
- 템플릿으로 저장 244
- 페이지 속성 248
- 폴더 선택 234
- 표를 레이어로 변환 230
- 프레임에서 열기 241
- 환경 설정 108, 111, 113
- 데이터 관리자 310
- 데이터 소스
 - ColdFusion 61
 - ODBC 63
- 데이터베이스
 - 데이터베이스 API 53
 - 뷰 73
 - 액세스 함수 65
 - 연결 대화 상자 API 77
 - 연결 유형 정의 파일 81
 - 연결 함수 53
- 데이터베이스 API
 - MMDB.deleteConnection() 54
 - MMDB.getColdFusionDsnList() 54
 - MMDB.getColumnAndTypeList() 65
 - MMDB.getColumnList() 66
 - MMDB.getColumns() 67
 - MMDB.getColumnsOfTable() 67
 - MMDB.getConnection() 55
 - MMDB.getConnectionList() 56
 - MMDB.getConnectionName() 56
 - MMDB.getConnectionString() 57
 - MMDB.getDriverName() 57
 - MMDB.getLocalDsnList() 58
 - MMDB.getPassword() 58
 - MMDB.getPrimaryKeys() 68
 - MMDB.getProcedures() 68
 - MMDB.getRdsPassword() 58
 - MMDB.getRdsUserName() 59
 - MMDB.getRemoteDsnList() 59
 - MMDB.getRuntimeConnectionType() 59
 - MMDB.getSPColumnList() 69
 - MMDB.getSPColumnListNamedParams() 70
 - MMDB.getSPParameters() 71
 - MMDB.getSPParamsAsString() 72
 - MMDB.getTables() 72
 - MMDB.getUserName() 60
 - MMDB.getViews() 73
 - MMDB.hasConnectionWithName() 60
 - MMDB.needToPromptForRdsInfo() 61
 - MMDB.needToRefreshColdFusionDsnList() 61
 - MMDB.popupConnection() 61
 - MMDB.setRdsPassword() 62
 - MMDB.setRdsUserName() 62
 - MMDB.showColdFusionAdmin() 63
 - MMDB.showConnectionMgrDialog() 63
 - MMDB.showOdbcDialog() 63
 - MMDB.showRdsUserDialog() 64
 - MMDB.showRestrictDialog() 64
 - MMDB.showResultset() 74
 - MMDB.showSPResultset() 74
 - MMDB.showSPResultsetNamedParams() 75
 - MMDB.testConnection() 64
 - 액세스 함수 65
 - 연결 함수 53
 - 정보 53
- 데이터베이스 연결
 - 사용자 이름 60
 - 삭제 54
 - 암호 58
 - 존재 여부 테스트 60
- 데이터베이스 연결 API. 데이터베이스 연결 대화 상자 API 참조
- 데이터베이스 연결 개요 76
- 데이터베이스 연결 대화 상자 API
 - applyConnection() 79
 - findConnection() 78
 - inspectConnection() 79
- 정보 77
- 정의 파일 81
- 포함 파일, 생성됨 80
- 데이터베이스 연결 유형 정의 파일 81
- 데이터베이스 연결(MMDB) 함수 53
- 도움말 파일, 열기 106
- 동적 데이터 대화 상자 112
- 동적 문서 함수 308
- 동적 텍스트 대화 상자 112
- 되돌리기
 - 문서 242
 - 체크 아웃 92
- 드라이버 이름 57
- 드라이버, JDBC 57
- 들여쓰기 148
- 디자인 노트
 - C API 27
 - 길이 94
 - 소스 제어 시스템 95
 - 파일 구조 22
- 디자인 노트 키 94
- 디자인 노트 파일
 - 닫기 27
 - 사이트 루트 24, 30
 - 열기 26, 31
 - 저장 22
 - 키 24, 28
 - 키 가져오기 23
 - 키 제거 26, 32
 - 키/값 쌍 23
 - 키/값 쌍 만들기 26, 32
 - 키/값 쌍의 수 29
- 디자인 노트 파일에서 키 제거 32

디자인 노트 함수

MMNotes.close() 22
MMNotes.filePathToLocalURL() 23
MMNotes.get() 23
MMNotes.getKeyCount() 23
MMNotes.getKeys() 24
MMNotes.getSiteRootForFile() 24
MMNotes.getVersionName() 25
MMNotes.getVersionNum() 25
MMNotes.localURLToFilePath() 25

디자인 뷰

표시 171
표시/숨기기 설정 172

디자인 함수 329

ㄹ

라이브 데이터 함수 312
라이브 뷰 함수 316
라이브러리 및 템플릿 함수 287
런타임 연결 유형 60
레이아웃 뷰 358, 362
레이아웃 뷰 함수 357
레이아웃 환경 함수 352
레이어 351
HTML 태그 351
이동 350
정렬 349
크기 조정 350, 351

레이어 함수 349

레이어를 표로, 변환 229

렌더링

스타일 340

로그인 정보, RDS 61, 64

로컬 드라이브 경로

파일 URL로 변환 23, 27

파일 URL에서 변환 25

로컬 루트 폴더 254

로컬 파일 경로, 사이트 상대 URI로 변환 198, 199

루트 폴더 이름 85

링크 확인 135

ㅁ

만들기

XML 파일 236, 237

결과 윈도우 140

문서 235

폴더 8, 87

맞춤법 검사 245

맞춤법, 검사 245

매개 변수, 저장 프로시저 72

맨 앞

Dreamweaver 가져오기 33

Fireworks 가져오기 33

메뉴

가져오기 97

삽입 막대 126

새로 고침 135

업데이트 134, 135

연결 96

올리기 97

체크 아웃 97

메뉴 함수

라이브 데이터 함수로 모방 312

메뉴 최적화 및 새로 고침 134

명령 메뉴 230

메뉴 항목

다시 실행 121

삭제 99

새 폴더 98

실행 취소 121

이름 변경 99

체크 아웃 취소 98

체크 인 98

명령

기록된 119

실행 230

명령 메뉴 함수 230

명령으로 저장 대화 상자 124

목록

브라우저 103

열려 있는 문서 174

최근에 사용한 파일 239

편집기 103

문서

닫기 234

되돌리기 242

만들기 235, 240

열기 239, 240

유효성 검사 246

저장 242, 243

처음 133

문서 윈도우 계단식 배열 173

문서 유효성 검사 246

문서 정보 237

문서 함수 229

문서 함수, 전역 245

문서의 처음 133

문자열

파일 내용 12

파일에 쓰기 14

ㅂ

바탕 이미지

불투명도 353

정렬 356

백스페이스 키, 누르기 129

버전

Fireworks 36

Flash MX 108

소스 제어 시스템 84

버전 번호, MMNotes 공유 라이브러리 25, 30

버전 이름, MMNotes 공유 라이브러리 25, 30

변환

로컬 드라이브 경로를 파일 URL로 23, 27

로컬 파일 경로를 사이트 상대 URI로 198

백분율에서 픽셀로 373

사이트 상대 URI를 로컬 파일 경로로 199

파일 URL을 로컬 드라이브 경로로 25, 31

픽셀에서 백분율로 373

변환 함수 229, 263

보고 함수 197

보고서

결과 패널 135

보조 브라우저 105

복사

선택 영역 283

작업 내역 단계 121

파일 7

부동 패널 함수 171

부동 패널, 정렬 357

- 불투명도, 바탕 이미지 353
- 붙여넣기 112
- 뷰 73
 - 선택 172
 - 표시 176
 - 확인 171
- 뷰 비율 363
- 뷰 테이블 73
- 브라우저
 - URL 열기 102
 - 기본 105
 - 대상 135
 - 목록 103
 - 문서 확인 245
 - 보조 105
- 브라우저 호환성 확인 함수 300
- 브라우저에서 문서 확인 245
- 블록
 - ID 359, 361
 - 색상 표시 358, 360
 - 테두리 359, 361
- 비교, 원격 및 로컬 파일 96
- 비헤이비어 함수 275
 - 서버 322
- 人
- 사용되지 않는 함수 4
- 사용자 이름 60
 - RDS 59, 62, 64
 - 이름 체크 아웃 91
- 사용자, 파일 체크 아웃 93
- 사이트
 - 로컬 루트 폴더 254
 - 사이트 정보 저장 200
 - 삭제된 89
 - 이름 변경 90
 - 파일 비교 응용 프로그램 실행 200
- 사이트 루트, 디자인 노트 파일 24, 30
- 사이트 보고서 136
- 사이트 이름 변경 90
- 사이트 패널 선택 함수 198
- 사이트 함수 197, 198
- 사이트, 모든 사이트에 대한 정보 200

- 삭제
 - 데이터베이스 연결 54
 - 선택 영역 392
 - 스타일 338
 - 폴더 16
- 삽입
 - 문서에 문자열 112
 - 태그 113
- 삽입 막대
 - 객체 새로 고침 126
 - 메뉴 126
- 삽입 페이지 132
- 삽입점
 - 다음 단락의 처음 131
 - 다음 단어 131
 - 문서의 끝 130
 - 문서의 처음 133
 - 이동 127, 128, 129
 - 이전 단락 132
 - 이전 단어 132
 - 한 페이지 아래 131
 - 행의 끝 130
 - 행의 처음 133
- 상자 모델, 색상 표시 360
- 새 문서 240
- 새로 고침 386
 - 삽입 막대의 객체 126
- 색상 표시
 - div 358, 360
 - 레이아웃 블록 배경 346
 - 블록 358, 360
 - 상자 모델 345, 360, 362
 - 안내선 368
 - 코드 382, 386
- 생성된 포함 파일
 - ASP JavaScript 80
- 서버
 - 구성 요소 함수 308
 - 디버깅 145
 - 비헤이비어 함수 322
- 선택 영역 171
- 삭제 392
- 선택 함수
 - 사이트 패널 198
 - 열린 문서 255

- 선택, 작업 내역 단계 124
- 선택하여 붙여넣기 대화 상자 112
- 세로 셀, 나누기 374
- 소스 유효성 검사 135
- 소스 제어 및 Dreamweaver
 - 통합 83
- 소스 제어 시스템 91
 - 디자인 노트 95
 - 디자인 노트 길이 94
 - 디자인 노트 키 94
 - 루트 폴더 이름 85
 - 루트 폴더 이름 길이 85
 - 버전 84
 - 사이트 이름 변경 90
 - 삭제된 사이트 89
 - 새 기능 90
 - 여러 파일 99, 100, 101
 - 연결 84, 89
 - 연결 테스트 85
 - 연결 해제 84
 - 오류 메시지 94
 - 원격 파일 96
 - 이름 84
 - 이름 체크 아웃 91
 - 전달된 폴더 86
 - 주석 추가 99
 - 파일 87
 - 파일 삭제 88
 - 파일 올리기 87
 - 파일 이름 변경 88
 - 파일 존재 여부 테스트 88
 - 폴더 만들기 87
 - 폴더 항목 86
- 소스 제어 시스템 기능
 - 추가 83
- 소스 제어 시스템 연결 해제 84
- 소스 제어 시스템에 연결 84
- 소스 제어 통합 API
 - SCS_AfterGet() 100
 - SCS_AfterPut() 101
 - SCS_BeforeGet() 99
 - SCS_BeforePut() 100
 - SCS_canCheckin() 98
 - SCS_canCheckout() 97

- SCS_canConnect() 96
- SCS_canDelete() 99
- SCS_canGet() 97
- SCS_canNewFolder() 98
- SCS_canPut() 97
- SCS_canRename() 99
- SCS_CanUndoCheckout() 98
- SCS_Checkin() 91
- SCS_Checkout() 92
- SCS_Connect() 84
- SCS_Delete() 88
- SCS_Disconnect() 84
- SCS_Get() 87
- SCS_GetAgentInfo() 84
- SCS_GetCheckoutName() 91
- SCS_GetConnectionInfo() 89
- SCS_GetDesignNotes() 95
- SCS_GetErrorMessage() 94
- SCS_GetErrorMessageLength() 93
- SCS_GetFileCheckoutList() 93
- SCS_GetFolderList() 86
- SCS_GetFolderListLength() 86
- SCS_GetMaxNoteLength() 94
- SCS_GetNewFeatures() 90
- SCS_GetNoteCount() 94
- SCS_GetNumCheckedOut() 93
- SCS_GetNumNewFeatures() 90
- SCS_GetRootFolder() 85
- SCS_GetRootFolderLength() 85
- SCS_IsConnected() 85
- SCS_IsRemoteNewer() 96
- SCS_ItemExists() 88
- SCS_NewFolder() 87
- SCS_Put() 87
- SCS_Rename() 88
- SCS_SetDesignNotes() 95
- SCS_SiteDeleted() 89
- SCS_SiteRenamed() 90
- SCS_UndoCheckout() 92
- 정보 83
- 소스 제어 통합 API 선택적 함수 89
- 소음 110
- 속성
 - snippet 태그 293
 - 가져오기 9
 - 파일, 설정 13
 - 수, 체크 아웃한 파일 93
 - 수직 분할 뷰 함수 183
 - 숨김 파일 13
 - 스마트 오브젝트 49
 - 스마트 오브젝트 API 49
 - 스타일
 - 렌더링 340, 344
 - 목록 342
 - 복제 339
 - 삭제 338
 - 이름 가져오기 341
 - 이름 변경 343
 - 적용 331, 337
 - 제거 334
 - 스타일 복제 339
 - 스타일 시트 338
 - 스타일 적용 331
 - 시각 도구 148, 153, 345, 346
 - 레이아웃 블록 배경 333, 335, 346
 - 레이아웃 블록 상자 모델 333, 336, 345
 - 레이아웃 블록 외곽선 333, 336
 - 시간
 - 생성된 파일 10
 - 수정된 파일 9
 - 시스템 경고음 110
 - 실행 취소 118, 120, 125
 - 다시 실행 117
 - 상태 123
 - 실행 취소, 파일 체크 아웃 92
 - 쓰기 가능한 파일 13
-
- 안내선
 - 작업 366
 - 잠금 369
- 안내선 함수 366
- 암호
 - RDS 58, 62, 64
 - 데이터베이스 연결 58
- 양식, 게시 20, 21
- 업데이트
 - 디자인 노트 파일에서 키/값 쌍 26
 - 메뉴 134, 135
- 에셋 패널(팔레트) 함수 268
- 여러 파일 99, 100
- 연결 62
 - HTML 생성 79
 - JDBC 57
 - 검색 78
 - 목록 가져오기 56, 65
 - 소스 제어 시스템 85, 89
 - 정의 79
 - 특정 이름 가져오기 56
- 연결 개요 76
- 연결 객체 55
 - 속성 78
- 연결 관리자 56, 61, 63
- 연결 문자열 56, 57
 - 테스트 64
- 연결 유형
 - 런타임 60
 - 만들기 76
- 연결 이름 56
- 연결 정의 파일 81
- 연결 처리, 데이터베이스 53
- 열 66, 67
 - SQL SELECT 65
- 결과 윈도우 내의 폭 143
- 결과 집합 70
- 명령문에서 가져오기 65, 66
- 유형 65
- 이름 68
 - 저장 프로시저에서 가져오기 69, 70
 - 테이블에서 가져오기 67
- 열기
 - 도움말 파일 106
 - 디자인 노트 파일 26, 31
 - 문서 239, 240
 - 외부 텍스트 편집기에서 문서 108
 - 지정된 응용 프로그램으로 파일 107
 - 지정된 이미지 편집기로 파일 108
- 열려 있는 문서, 목록 174
- 오디오 경고 110

오류 메시지 93

소스 제어 시스템 93, 94

옵션, 객체 삽입시 대화 상자 보기 111

외곽선 속성 345, 346

외부 응용 프로그램 함수 102

외부 텍스트 편집기 104, 108

원격 파일 96

웹 페이지 내용 함수 268

위젯, Spry

삽입 함수 298

편집 함수 296

윈도우

계단식 배열 173

최소화 177

윈도우 최소화 177

윈도우 함수 171

유형, 열 65

유효성 검사기 메서드 136

응용 프로그램

선택 107

파일 열기 107

응용 프로그램 함수

bridge 통신 115

외부 102

전역 110

이동

레이어 350

삽입점 127, 128, 129

핫스팟 350

이름

소스 제어 시스템 84

열 68

체크 아웃 91

이름 변경

스타일 343

파일 88

이름 체크 아웃 91

이미지 맵 함수 349

이미지 편집기 108

이전 단락 132

이전 단어 132

인쇄 규칙 5

읽기 전용 파일 13

읽기, Flash 객체 파일 41

ㅈ

자동 들여쓰기 162

작업 내역 단계

복사 121

선택 124

재실행 123

해당하는 JavaScript 122

작업 내역 단계 재실행 123

작업 내역 패널 121

단계 122

작업 내역 패널 지우기 121

작업 내역 함수 117

작업 영역 API 함수

정보 117

잘못된 HTML 148, 150, 163

잠금, 안내선 369

재생

기록된 명령 119

플러그인 내용 353

플러그인 항목 354

저장

디자인 노트 파일 22

문서 242, 243

작업 내역 단계 124

저장 프로시저 70, 71, 74, 75

결과 표시 74, 75

매개 변수 72

매개 변수 가져오기 72

열 가져오기 69, 70

정보 65

전달된 폴더 내, 파일 86

전역 문서 함수 245

전역 응용 프로그램 함수 110

전환 함수 147

정렬

레이어 349

바탕 이미지 356

부동 패널 357

핫스팟 350

정리, XHTML 문서 231

정보 대화 상자 111

정보 막대 함수 181

정의 파일, 연결 유형 81

제거

ID 문자열 125

스타일 334

제거된 함수 4

제한 대화 상자 64

존재, 데이터베이스 연결 60

주석, 적용 193

줄 바꿈 159, 162, 164

중단

기록 120

플러그인 내용 356

즐거찾기 목록

제거 273

추가 268, 269

ㅊ

최적화 세션, Fireworks 35

ㅋ

코드 관리자

구문 색상 표시 162

자동 들여쓰기 162

잘못된 HTML 161, 163

줄 바꿈 162, 164

행 번호 161

코드 단편 패널 함수 292

코드 뷰 171, 172

구문 색상 표시 152

자동 들여쓰기 148

잘못된 HTML 148

전환 339

줄 바꿈 153, 159

행 번호 151, 157

코드 색상 표시 386

코드 함수

코드 단편 패널 292

코드 뷰 409

코드 힌트 및 색상 382

크기

Flash 내용 40

파일 11

크기 조정

레이어 350, 351

핫스팟 351

클립보드 함수 283

키

Delete 129

Page Down 131

Page Up 132

값 가져오기 28

기본 68

디자인 노트 23

디자인 노트 파일 24, 28

디자인 노트 파일에서 제거 26

목록 29

백스페이스 129

키 코드, 문자로 변환 134

키/값 쌍

디자인 노트 파일 23

디자인 노트 파일에서 만들기 26

만들기 32

수 29

키보드 함수 127

E

태그

font 392

레이어 351

삽입 113

태그 선택기 대화 상자 113

태그 편집기 및 태그 라이브러리 함수 426

테두리 359

div 361

블록 361

테스트, 연결 문자열 64

테이블

데이터베이스 테이블 72

목록 가져오기 72

열 67

열 가져오기 67

텍스트

가져오기 118

게시 21

편집 작업 118

텍스트 편집기, 외부 104

템플릿 및 라이브러리 함수 287

툴바 숨김 159

툴바 표시 159

툴바 함수 165

툴바, 표시 159

표

파일

connection_includefile.edml 81

결과 윈도우 144

기본 편집기 105

내용 가져오기 19, 20

내용을 문자열로 읽기 12

도움말 106

만들기(HTML 파일) 235

만들기(XHTML 파일) 235

만들기(XML 파일) 236

만들기(비 HTML 파일) 14

문자열 쓰기 14

복사 7

비교 199

삭제 17, 88

생성된 시간 10

소스 제어 시스템 86, 87

속성 13

속성 가져오기 9

수정된 시간 9

쓰기 14

올리기 87, 101

이름 변경 88

읽기 12

저장 18, 19

전달된 폴더 내 86

제거 13

존재 여부 테스트 8, 88

지정된 응용 프로그램으로 열기 107

지정된 이미지 편집기로 열기 108

처리 145

체크 아웃 92, 93

체크 아웃 실행 취소 92

체크 아웃한 수 93

체크 인 91, 101

최근에 사용 239

코드 단편 292

크기 11

포함, 생성된 80

파일 I/O API

DWfile.copy() 7

DWfile.createFolder() 8

DWfile.exists() 8

DWfile.getAttributes() 9

DWfile.getCreationDate() 10

DWfile.getCreationDateObj() 10

DWfile.getModificationDate() 9

DWfile.getModificationDateObj() 11

DWfile.getSize() 11

DWfile.listFolder() 11

DWfile.read() 12

DWfile.remove() 13

DWfile.setAttributes() 13

DWfile.write() 14

정보 7

파일 URL

로컬 드라이브 경로로 변환 25, 31

로컬 파일 경로로 변환 23

파일 올리기 87, 101

파일 조작 함수 231

파일 처리 144, 145

파일 체크 아웃 92

수 93

실행 취소 92

파일 체크 인 91, 101

페이지 내용 함수 268

편집기, 목록 103

포커스가 있는 문서 선택 영역 171

포함 파일

ASP 및 JavaScript 80

ColdFusion 80

생성된 80

연결 유형 정의 81

폴더

_mmServerScripts 16

Configuration/Temp 17, 18, 19

구성 7

내용 12

만들기 8, 87

삭제 88

소스 제어 시스템 86, 87

소스 제어 시스템 체크 인/체크 아웃 92

속성 가져오기 9

- 올리기 87
- 제거 16
- 존재 여부 테스트 88
- 체크 인 91
- 표
 - 레이어로 변환 229
- 표 편집 함수 372
- 표시 171
 - 시각 도구 148
 - 키 코드 134
- 표시 파일 13
- 표준 뷰 358
- 프레임
 - 목록 348
 - 분할 349
- 프레임 및 프레임세트 함수 347
- 프레임 분할 349
- 프레임세트 348
 - 저장 348
- 프로시저, 해당 이름의 연결 68
- 플러그인 내용
 - 재생 353
 - 중단 356
- 플러그인 항목, 재생 354
- ㅎ
 - 한 페이지 위 132
- 함수
 - CS4에서 사용되지 않음 4
 - CSS 레이아웃 329
 - Dreamweaver CS4에 새로 추가 2
 - Extension Data Manager 310
 - Spry 위젯 삽입 298
 - Spry 위젯 편집 296
 - XSLT 265
 - 객체 삽입 125
 - 결과 윈도우 135
 - 경로 252
 - 데이터 소스 309
 - 라이브 데이터 312
 - 라이브러리 및 템플릿 287
 - 레이아웃 뷰 357
 - 레이아웃 환경 352
 - 레이어 및 이미지 맵 349

- 메뉴 134
- 명령 230
- 문자열 조작 260
- 변환 229, 263
- 보고 197
- 브라우저 호환성 확인 300
- 비헤이비어 275
- 사이트 198
- 서버 구성 요소 308
- 서버 모델 323
- 서버 비헤이비어 322
- 선택 255
- 안내선 366
- 에셋 패널 268
- 외부 응용 프로그램 102
- 윈도우 171
- 인쇄 406
- 일반 편집 391
- 작업 내역 117
- 전역 문서 245
- 전역 응용 프로그램 110
- 전환 147
- 찾기/바꾸기 386
- 코드 382
- 코드 단편 패널 292
- 코드 뷰 409
- 코드 뷰 톨바 191
- 코드 축소 185
- 퀵 태그 편집기 407
- 클립보드 283
- 키보드 127
- 태그 라이브러리 426
- 태그 편집기 426
- 툴바 165
- 파일 내용 전달 20
- 파일 조작 231
- 표 편집 372
- 프레임 및 프레임세트 347
- 확대/축소 363
- 활성자 431
- 핫스팟
 - 이동 350
 - 정렬 350
 - 크기 조정 350, 351
- 핫스팟 함수 349

- 해당 이름의 연결 57
 - 프로시저 68
- 해당 이름의 프로시저 71
- 해당하는 JavaScript, 작업 내역 단계 122
- 행 번호 151, 157, 161, 163
- 행, 처음 133
- 확대/축소 363
- 확대/축소 함수 363
- 환경 설정 대화 상자 113
- 활성자 96
- 활성자 함수, 정보 431
- 힌트, 코드 382