

# Scripting Guide

## ADOBE® FRAMEMAKER® 11



© 2014 Adobe Systems Incorporated. All rights reserved.

Adobe® FrameMaker® 12 Scripting Guide

If this guide is distributed with software that includes an end-user agreement, this guide, as well as the software described in it, is furnished under license and may be used or copied only in accordance with the terms of such license. Except as permitted by any such license, no part of this guide may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, recording, or otherwise, without the prior written permission of Adobe Systems Incorporated. Please note that the content in this guide is protected under copyright law even if it is not distributed with software that includes an end-user license agreement.

The content of this guide is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in the informational content contained in this guide.

Please remember that existing artwork or images that you may want to include in your project may be protected under copyright law. The unauthorized incorporation of such material into your new work could be a violation of the rights of the copyright owner. Please be sure to obtain any permission required from the copyright owner.

Any references to company names in sample templates are for demonstration purposes only and are not intended to refer to any actual organization.

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/>

Adobe, the Adobe logo, Adobe AIR, Adobe Captivate, Adobe Type Manager, Acrobat, Distiller, Flash, FrameMaker, Illustrator, PageMaker, Photoshop, PostScript, Reader, RoboHelp, and RoboScreenCapture are trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Microsoft, Windows, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries. SVG is a trademark of the World Wide Web Consortium; marks of the W3C are registered and held by its host institutions MIT, INRIA, and Keio. Helvetica is a registered trademark of is a trademark of Heidelberger Druckmaschinen AG exclusively licensed through Linotype Library GmbH, and may be registered in certain jurisdictions. ITC Zapf Dingbats is a trademark of International Typeface Corporation. Times New Roman is a registered trademark of The Monotype Corporation registered in the U.S. Patent and Trademark Office and may be registered in certain other jurisdictions. All other trademarks are the property of their respective owners.

This product contains either BSAFE and/or TIPEM software by RSA Data Security, Inc.

This product contains color data and/or the Licensed Trademark of The Focoltone Colour System.

PANTONE® Colors displayed in the software application or in the user documentation may not match PANTONE-identified standards. Consult current PANTONE Color Publications for accurate color. PANTONE® and other Pantone, Inc. trademarks are property of Pantone, Inc. © Pantone, Inc. 2003. Pantone, Inc. is the copyright owner of color data and/or software which are licensed to Adobe Systems Incorporated to distribute for use only in combination with Adobe FrameMaker. PANTONE Color Data and/or Software shall not be copied onto another disk or into memory unless as part of the execution of Adobe FrameMaker software.

Software is produced under Dainippon Ink and Chemicals Inc.'s copyrights of color-data-base derived from Sample Books.

This product contains ImageStream® Graphics and Presentation Filters Copyright ©1991-1996 Inso Corporation and/or Outside In® Viewer Technology ©1992-1996 Inso Corporation. All Rights Reserved.

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

Portions contributed by Gilles Vollant.

Certain Spelling portions of this product is based on Proximity Linguistic Technology. ©Copyright 1990 Merriam-Webster Inc. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2003 Franklin Electronic Publishers Inc. ©Copyright 2003 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. Legal Supplement ©Copyright 1990/1994 Merriam-Webster Inc./Franklin Electronic Publishers Inc. ©Copyright 1994 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990/1994 Merriam-Webster Inc./Franklin Electronic Publishers Inc. ©Copyright 1997 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA ©Copyright 1990 Merriam-Webster Inc. ©Copyright 1993 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 Franklin Electronic Publishers Inc. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1991 Dr. Lluís de Yzaguirre I Maura ©Copyright 1991 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Munksgaard International Publishers Ltd. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Van Dale Lexicografie bv ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1995 Van Dale Lexicografie bv ©Copyright 1996 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 IDE a.s. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1992 Hachette/Franklin Electronic Publishers Inc. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1991 Text & Satz Datentechnik ©Copyright 1991 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 Bertelsmann Lexikon Verlag ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 MorphoLogic Inc. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 William Collins Sons & Co. Ltd. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1993-95 Russicon Company Ltd. ©Copyright 1995 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 2004 IDE a.s. ©Copyright 2004 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. The Hyphenation portion of this product is based on Proximity Linguistic Technology. ©Copyright 2003 Franklin Electronic Publishers Inc. ©Copyright 2003 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1984 William Collins Sons & Co. Ltd. ©Copyright 1988 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Munksgaard International Publishers Ltd. ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1997 Van Dale Lexicografie bv ©Copyright 1997 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1984 Editions Fernand Nathan ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1983 S Fischer Verlag ©Copyright 1997 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1989 Zanichelli ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA.

©Copyright 1989 IDE a.s. ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1990 Espasa-Calpe ©Copyright 1990 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA. ©Copyright 1989 C.A. Stromberg AB. ©Copyright 1989 All rights reserved. Proximity Technology A Division of Franklin Electronic Publishers, Inc. Burlington, New Jersey USA.

Portions of Adobe Acrobat include technology used under license from Autonomy, and are copyrighted.

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. 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.

# Contents

## Chapter 1: Overview

About this document .....	1
Scripting support in FrameMaker .....	2
Creating Scripts .....	3
Accessing ESTK .....	3
Using script library .....	4
Using object model viewer .....	5
ExtendScript use cases .....	5

## Chapter 2: What's new in ESTK 12

Support MathML .....	6
CMS connector framework .....	6
Support for background color in objects .....	6
Support for paragraph background color in objects .....	6
New properties .....	6

## Chapter 3: Sample ESTK scripts

Adding text to a document and enabling change bars .....	8
Changing the table properties in a document .....	8

## Chapter 4: Differences between scripts and FDK

Naming differences between scripts and FDK .....	10
Notifications .....	13
Menus and Commands .....	14

## Chapter 5: Object Reference

AFrame .....	17
Arc .....	21
AttrCondExpr .....	24
Attribute .....	24
AttributeDef .....	25
AttributeDefs .....	26
AttributeEx .....	27
Attributes .....	27
AttributesEx .....	27
BodyPage .....	28
Book .....	29
BookComponent .....	38
Cell .....	47
CharFmt .....	49
CMSCheckinParam .....	52
CMSDeleteParam .....	53
CMSInfo .....	53
CMSMenuItem .....	53

**Contents**

CMSProperty	54
CMSProperties	54
CMSRegistration	54
CMSResult	55
CMSSession	55
Color	56
CombinedFont	57
CombinedFontDefn	58
CombinedFonts	58
Command	59
CompareRet	63
CondFmt	64
Constants	65
Doc	253
Element	287
ElementCatalogEntries	292
ElementCatalogEntry	292
ElementDef	292
ElementLoc	294
ElementRange	295
Ellipse	295
FCodes	298
FilterArgs	360
Flow	360
FmtChangelist	361
FmtRule	367
FmtRuleClause	367
Fn	368
Font	369
Fonts	369
Group	370
HiddenPage	372
GraphicsFmt	373
Inset methods	374
IdValuePair	374
IdValuePairs	374
Inset	374
Ints	378
KeyCatalog	379
Line	379
Marker	382
MarkerType	382
MasterPage	383
Math	383
MathML	387
Menu	390

**Contents**

MenuItemSeparator .....	390
Metrics .....	391
Pgf .....	391
PgfFmt .....	397
Point .....	402
Points .....	402
Polygon .....	403
Polyline .....	406
PropIdent .....	408
PropVal .....	409
PropVals .....	409
Rectangle .....	409
RefPage .....	412
RoundRect .....	413
Row .....	415
Rubi .....	417
RulingFmt .....	417
Strings .....	418
SubCol .....	418
Tab .....	419
Tabs .....	419
Tbl .....	420
TblFmt .....	423
TextFrame .....	425
TextItem .....	429
TextItems .....	431
TextLine .....	432
TextLoc .....	435
TextRange .....	435
TiApiClient .....	436
TiFlow .....	437
TiText .....	438
TiTextTable .....	439
TypedVal .....	440
TypedVals .....	440
UBytes .....	441
UInts .....	441
UnanchoredFrame .....	441
Var .....	444
VarFmt .....	445
XRef .....	446
XRefFmt .....	447
app .....	447

**Contents****Chapter 6: Function Summary**

AFrame	453
Arc	456
AttrCondExpr	457
Attribute	458
AttributeDef	459
AttributeDefs	460
Attributes	461
AttributesEx	462
BodyPage	463
Book	465
BookComponent	490
Cell	491
CharFmt	497
CMSObject	499
CMSProperties	508
CMSRegistration	509
CMSSession	511
Color	513
CombinedFont	515
CombinedFontDefn	515
CombinedFonts	517
Command	518
CompareRet	520
CondFmt	520
Doc	522
Element	600
ElementCatalogEntries	603
ElementCatalogEntry	604
ElementDef	604
ElementLoc	613
ElementRange	614
Ellipse	614
FilterArgs	615
Flow	616
FmtChangeList	618
FmtRule	619
FmtRuleClause	621
Fn	623
Font	625
Fonts	626
Group	627
HiddenPage	628
IdValuePairs	630
Inset	631
Ints	637

**Contents**

Iterator	638
KeyCatalog	640
Line	646
Marker	647
MarkerType	649
MasterPage	650
Math	652
MathML	653
Menu	654
MenuItemSeparator	661
Metrics	663
Pgf	664
PgfFmt	665
Point	667
Points	667
Polygon	668
Polyline	670
PropIdent	671
PropVal	671
PropVals	672
Rectangle	673
RefPage	674
RoundRect	676
Row	677
Rubi	679
RulingFmt	682
Strings	683
SubCol	684
Tab	686
Tabs	686
Tbl	688
TblFmt	692
TextFrame	692
TextItems	694
TextLine	695
TextLoc	697
TextRange	697
TiApiClient	698
TiFlow	700
TiText	701
TiTextTable	702
TypedVals	703
UBytes	704
UInts	706
UnanchoredFrame	707
Var	708



VarFmt .....	710
XRef .....	711
XRefFmt .....	713
app .....	714
global .....	720
 <b>Chapter 7: CMS Connector Framework</b>	
CMS API Data Structures and Enum Constants .....	765
Error Codes .....	767
CMS API functions .....	768
APIs to automate CMS connector functionality .....	769

# Chapter 1: Overview

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

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

## About this document

### Naming conventions

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

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

### Typographic conventions

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

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

## Scripting support in FrameMaker

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

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

### Development and debugging tools in ExtendScript toolkit

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

### ExtendScript capabilities

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

- **User interface development tools:** The ScriptUI module defines Window objects that represent platform-specific windows and various control elements, such as buttons and static text. Resource specification allows you to create user-interface elements. Using the event-handling callback functions, you can define the behavior of your user-interaction controls. ScriptUI provides a rich set of containers (such as Group, Panel, and TabbedPanel) and user interface controls (such as Button, Image, Scrollbar, Treeview, and FlashPlayer). ScriptUI works with the ExtendScript JavaScript interpreter to provide JavaScript programs with the ability to create and interact with user interface elements. The Graphic customization objects provide the ability to customize the appearance of user-interface controls before they are drawn. For details, see Chapter 4, “User-Interface Tools” in [JavaScript Tools Guide](#).
- **Extensions:**
  - ExtendScript offers tools for communicating with other computers or the Internet using standard protocols. The Socket object supports low-level TCP connections. For details, see Chapter 6, “External Communication Tools” in [JavaScript Tools Guide](#).
  - Adobe ExtendScript defines classes that simplify cross-platform file-system access. These classes are available to all applications that support a JavaScript interface. For details, see Chapter 3, “File System Access” in [JavaScript Tools Guide](#).
  - You can extend the JavaScript DOM for an application. To extend the JavaScript DOM, write a C or C++ shared library, compile it for the relevant platform, and load it into JavaScript as an ExternalObject instance. A shared library is implemented by a DLL in Windows, a bundle or framework in Mac OS, or a SharedObject in UNIX. For details, see Chapter 7, “Integrating External Libraries” in [JavaScript Tools Guide](#).
- **Inter-application communication and messaging:** ExtendScript provides a common scripting environment for all Adobe JavaScript-enabled applications. The cross-DOM API allows implementation of functions, such as open files and execute scripts, that are common across message-enabled applications. Using simple syntax you can implement application-specific functionality for message-enabled application. The BridgeTalk class of the messaging API has globally available static properties and functions that provide access to environmental information relevant for communication between applications. The error codes defined by the interapplication messaging protocol are compatible with the ExtendScript error codes. For details, see Chapter 5, “Interapplication Communication with Scripts” in [JavaScript Tools Guide](#)

## Creating Scripts

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

## Accessing ESTK

### Connect ESTK to FrameMaker

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

- 1 In the ExtendScript Toolkit, select **Adobe FrameMaker <Version#>** in the pop-up menu.
- 2 Click .

If FrameMaker is not already running, a message appears: “Target Adobe FrameMaker <Version#> is not running. Do you want to launch Adobe FrameMaker <Version#>?”

- 3 Click **Yes**.

FrameMaker is launched and connected to ESTK.

### To write, edit, and run scripts

In FrameMaker, do the following:

- 1 Click File > Script > New Script.

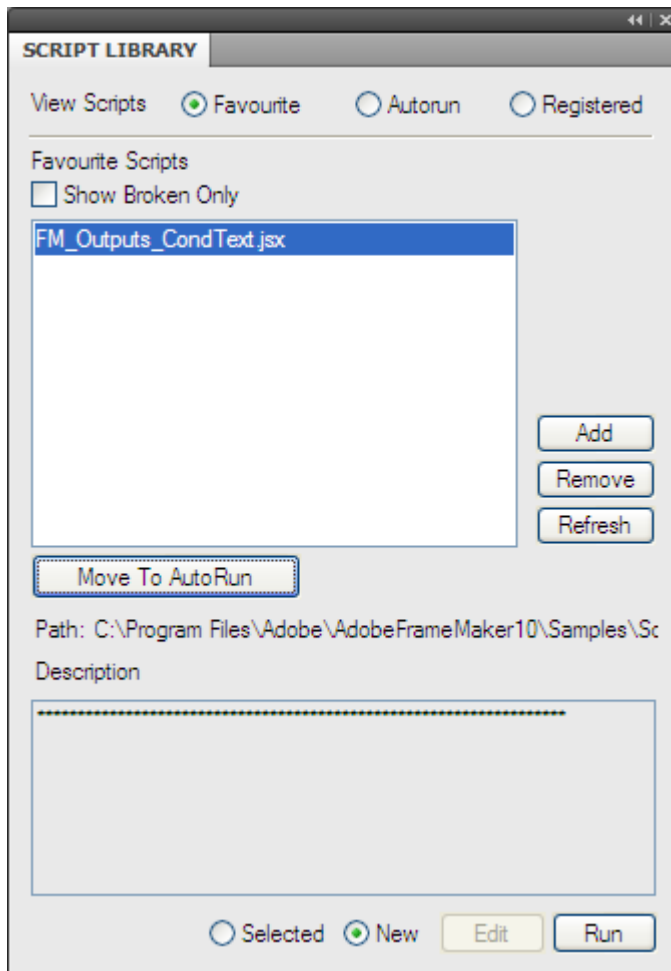
The ESTK editor opens.

- 2 Write the script in the editor.
- 3 To run the script, click the Play button.

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

## Using script library

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

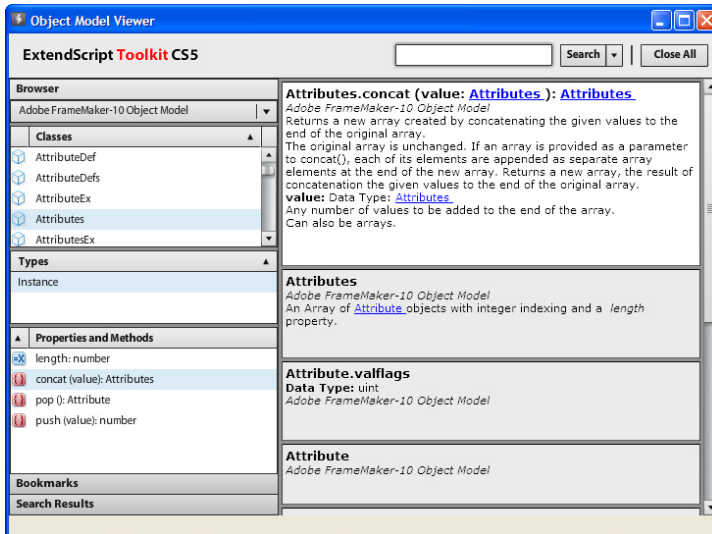


Using the script catalog, you can manage your existing scripts:

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

## Using object model viewer

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



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

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

## ExtendScript use cases

Some of the ExtendScript use cases can be:

- **Scripts to automate repetitive tasks:** You can write utilities to automate repetitive tasks, such as change alignment of all the tables in a FrameMaker document.
- **Notification-based scripts:** You can write scripts that are triggered on FrameMaker events such as such as opening, printing, or saving a document.
- **Menu command scripts/scripts to extend FrameMaker functionality:** Using these scripts, you can add menus and commands in FrameMaker.
- **Startup scripts:** These scripts get executed when FrameMaker starts. You can set up scripts to run at startup by copying them at the following location: `<FMINSTALL_DIR>\startup`. Alternatively, you can use the Script Library dialog to register Autorun scripts.

# Chapter 2: What's new in ESTK 12

## Support MathML

ESTK 12 provides support for the MathML “[MathML](#)” on page 386 object. Using the ESTK, you can now create and work with MathML equations.

To create a new MathML object instance, use “[NewMathML](#)” on page 560.

## CMS connector framework

The “[CMSObject](#)” on page 499 now extends support for the new “[CMSShowPropertyUIWithTitle](#)” on page 507 method.

## Support for background color in objects

ESTK 12 supports background color:

- [FmtChangeList](#)

## Support for paragraph background color in objects

ESTK 12 supports background color:

- [FmtChangeList](#)
- [Pgf](#)
- [PgfFmt](#)

## New properties

ESTK 12 provides the following new object properties:

Object	Properties
<a href="#">Command</a>	<p><b>Product:</b> Specifies the Product (Structured, unstructured, XML Author) to add the command.</p> <p><b>View:</b> Specifies the View (WYSIWYG, Author, XML) to add the command.</p> <p><b>Window:</b> Specifies the Window (Doc, Book, Ditamap) to add the command.</p>
“ <a href="#">app</a> ” on page 447	<p><b>CharacterReferencesOnXMLExport:</b> If there are leading/trailing whitespaces (spaces and tabs) in elements, save the whitespaces as character references to enable round-tripping.</p> <p><b>UserSettingsDir:</b> Name of the FrameMaker app directory.</p> <p><b>ProductFamily:</b> The FrameMaker product family name (FrameMaker, FrameMaker XML Author)</p>





## Chapter 3: Sample ESTK scripts

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

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

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

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

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

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

```
var doc = app.ActiveDoc;
var firstPgf = doc.MainFlowInDoc.FirstTextFrameInFlow.FirstPgf;
var t1 = new TextLoc(firstPgf, 0);
doc.AddText (t1, "Hello");
doc.AutoChangeBars = 1;
```

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

### Changing the table properties in a document

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

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

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

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

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

# Chapter 4: Differences between scripts and FDK

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

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

## Naming differences between scripts and FDK

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

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

### Example 1:

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

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

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

**Explanation:**

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

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

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

The same example in FDK is as follows:

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

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

**Example 2:**

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

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

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

```
doc.UpdateVariables();
```

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

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

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

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

**Global Methods:**

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

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

Consider the following line of code for FDK:

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

The same code can be written in scripts as follows:

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

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

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

### Example 3:

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

For example, structure `F_PointT`:

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

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

See another FDK structure called `F_TextRangeT`:

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

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

Notice that `F_TextLocT` itself is another structure.

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

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

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

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

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

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

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

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

Scripts have advanced features such as notifications and commands.

## Notifications

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

As an example, consider the following script:

```
/*
    Sample script to demonstrate how to use notification mechanism of FrameMaker though
    ExtendScript
*/
/*
There are basically two parts to a notification script.
    1. Registering for a desired notification. (This is done by calling 'Notification()' with
    desired notifications as argument.)
    2. Implement a handler for these notifications in your script. ( Implement 'Notify()' in
    script)
*/
/*
Part 1. Registering for Notifications
    Notification()' is a global function and is available readily in scripting. (Similar to
    F_ApiNotification())
    The Syntax for Notification() is: Notification(int, int,.....,int, bool);
    it takes:
        a.) list of notifications(FA_Note_PreOpenDoc, FA_Note_PostOpenDoc etc.) and
        b.) param bool: true = register for provided notification list, false = unregister for provided
    notification list.
    (This parameter is optional and by default 'true')
    Note: for more details please see F_ApiNotification() in FDK reference guide.
*/
/*
    Notification(FA_Note_PreOpenDoc,true);
    This is same as Notification(FA_Note_PreSaveDoc)
    or you can register for more than one notification at once
    Notification(FA_Note_PostOpenDoc,FA_Note_PreSaveDoc);
    Here we register for the notification: FA_Note_PostOpenDoc - this notification is
    triggered just after a document is opened.
    Notification(FA_Note_PostOpenDoc,true);
*/
/*
Part 2: handler
    This can be done by implementing function 'Notify()' that are called by FrameMaker when any
```

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

The line of code:

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

registers this script for the FA\_Note\_PostOpenDoc notification. This notification is triggered **just after** any document is opened in FrameMaker.

The lines of code:

```
function Notify(note, object, sparm, iparam)
{
switch (note) {
case FA_Note_PostOpenDoc:
alert("Opened document " + sparm);
break;
}
}
```

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

## Menus and Commands

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

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

```
/*
    In this script, we add two menu items to FrameMaker Session and associate two different
    commands to these menu items
*/
// Main menu bar
/* First we get the object associated with Main menubar in FrameMaker*/
mMenu = app.GetNamedMenu("!MakerMainMenu") ;
// F_ApiGetNamedMenu(FV_SessionId, FO_Menu, (ConStringT)!MakerMainMenu)
/*
Defining new menu items
DefineAndAddMenu(name, label)
FDK equivalent : F_ApiDefineAndAddMenu(F_ObjHandleT toMenuId, StringT name, StringT label)
*/
nMenu1 = mMenu.DefineAndAddMenu("Menu1", "Menu1");
nMenu2 = mMenu.DefineAndAddMenu("Menu2", "Menu2");
// Adding command to menus
/*
DefineAndAddCommand(cmdID, name, label, shortcut)
FDK equivalent: F_ApiDefineAndAddCommand(IntT cmdID, F_ObjHandleT toMenuId, StringT name,
StringT label, StringT shortcut)
cmdID is a integer that will be passed to Command() function defined in your script. It should
be unique to the script.
*/
nMenu1.DefineAndAddCommand(1, "cmd1", "command1", "");
nMenu2.DefineAndAddCommand(2, "cmd2", "command2", "");
// Another way for creating a command and adding it to menu items.
//cmd3 = DefineCommand(3, "cmd3", "cmd3", "");
// FDK equivalent : F_ApiDefineCommand(IntT cmdID, StringT name, StringT label, StringT shortcut)
// add the command to different menu items
//nMenu1.AddCommandToMenu(cmd3);
// FDK equivalent : ApiAddCommandToMenu(F_ObjHandleT toMenuId, F_ObjHandleT commandId)
//nMenu2.AddCommandToMenu(cmd3);
// must call if script has been run through ESTK, redundant otherwise.
UpdateMenus();
// Call back equivalent to F_ApiCommand(command)
function Command(cmd) {
    switch(cmd) {
        case 1:
            alert("Command1 executed");
            break;
        case 2:
            alert("Command2 executed");
            break;
    }
}
```

Let us examine the script in detail:

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

2 The lines:

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

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



**3** The commands:

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

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

**4** Finally, the block of code:

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

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

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

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

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

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

# Chapter 5: Object Reference

## AFrame

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

Property name	Data Type	Description
AnchorType	number	Indicates the location where the frame is anchored. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_ANCHOR_INLINE (1)</li> <li>• Constants.FV_ANCHOR_TOP (2)</li> <li>• Constants.FV_ANCHOR_BELOW (3)</li> <li>• Constants.FV_ANCHOR_BOTTOM (4)</li> <li>• Constants.FV_ANCHOR_SUBCOL_LEFT (5)</li> <li>• Constants.FV_ANCHOR_SUBCOL_RIGHT (6)</li> <li>• Constants.FV_ANCHOR_SUBCOL_NEAREST (7)</li> <li>• Constants.FV_ANCHOR_SUBCOL_FARTHEST (8)</li> <li>• Constants.FV_ANCHOR_SUBCOL_INSIDE (9)</li> <li>• Constants.FV_ANCHOR_SUBCOL_OUTSIDE (10)</li> <li>• Constants.FV_ANCHOR_TEXTFRAME_LEFT (11)</li> <li>• Constants.FV_ANCHOR_TEXTFRAME_RIGHT (12)</li> <li>• Constants.FV_ANCHOR_TEXTFRAME_NEAREST (13)</li> <li>• Constants.FV_ANCHOR_TEXTFRAME_FARTHEST (14)</li> <li>• Constants.FV_ANCHOR_TEXTFRAME_INSIDE (15)</li> <li>• Constants.FV_ANCHOR_TEXTFRAME_OUTSIDE (16)</li> <li>• Constants.FV_ANCHOR_RUN_INTO_PARAGRAPH (17)</li> </ul>
BaselineOffset	number	Baseline Offset
Element	<a href="#">Element</a>	If the anchored frame is in a structured flow in a FrameMaker document, Element indicates the element containing the anchored frame.
InTextFrame	<a href="#">TextFrame</a>	Indicates the text frame in which the anchored frame is placed.
InTextObj	FMOBJECT	Indicates the column or text frame in which the anchored frame is placed.
NextAFrame	<a href="#">AFrame</a>	Contains the next anchored frame in the text frame.
PrevAFrame	<a href="#">AFrame</a>	Contains the previous anchored frame in the text frame.
SideOffset	number	Denotes the near side offset.
TextLoc	<a href="#">TextLoc</a>	Contains the location of the anchor symbol.
FirstGraphicInFrame	FMOBJECT	Contains the first object in the anchored frame.
LastGraphicInFrame	FMOBJECT	Contains the last object in the anchored frame.
Angle	number	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	number	Denotes the arrowhead base angle in degrees.

Property name	Data Type	Description
ArrowLength	number	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	number	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is False.
ArrowScaleHead	number	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	number	Denotes the arrowhead tip angle in degrees.
ArrowType	number	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	number	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	Color	Denotes the spot color.
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	number	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	number	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	number	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	Group	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	number	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	number	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.

Property name	Data Type	Description
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	number	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	number	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is – 216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	number	Distance of the object from the top of the parent frame (in inches) The permissible range is –216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FObject	Denotes the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Denotes the next selected graphic object in the document.
ObjectAttributes	Strings	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	number	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	number	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>

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

## AFrame methods

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

## Arc

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

Property name	Data Type	Description
ArrowTipAngle	number	Specifies the arrowhead tip angle in degrees.
ArrowType	number	Denotes the Arrowhead style. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_ARROW_STICK (0x01)</li> <li>• Constants.FV_ARROW_HOLLOW (0x02)</li> <li>• Constants.FV_ARROW_FILLED (0x03)</li> </ul>
BorderWidth	number	Specifies the border width in points. The permissible range is 0.015 point to 360 point.
Color	Color	Indicates the spot color.
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a Metrics object. The 0 <sup>th</sup> element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	number	Denotes the fill pattern. The patterns are denoted by numbers between 0 and 15.  The following constants are provided for the fill patterns 0, 7 and 15: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FMOBJECT	Specifies the frame containing the graphic object.
GraphicCantBeSelected	number	Set to 1 if the graphic object cannot be selected. Else, set to 0.
GraphicIsSelected	number	Set to 1 if the graphic object is selected, else set to 0.
GroupParent	Group	Indicates the group to which the object belongs. Anchored and unanchored frames do not have this property.
HeadArrow	number	Set to 1 if line has an arrowhead, set to 0 otherwise.
Height	number	Specifies the height of the object in points. The permissible range is from 0.125 pt to 3600 pt.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.

Property name	Data Type	Description
LineCap	number	Indicates the type of line end. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	number	Specifies the distance of the object from the left side of the parent, in inches. The permissible range is from -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set the LocX property for anchored frames.
LocY	number	Specifies the distance of the object from the top of the parent frame, in inches. The permissible range is from -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You CANNOT set the LocY property for anchored frames.
NextGraphicInDoc	FObject	Indicates the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Indicates the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Specifies the next selected graphic object in the document.
ObjectAttributes	Strings	Contains a list of strings, each string expressing an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	number	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_KNOCKOUT (0x00)</li> <li>Constants.FV_OVERPRINT (0x01)</li> <li>Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	number	Indicates the pen pattern, denoted by numbers between 0 and 7.  Constants are provided for some of the fill patterns: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Specifies the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Indicates the previous graphic object in the group.



Property name	Data Type	Description
Runaround	number	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_TR_NONE (0x01)</li> <li>• Constants.FV_TR_CONTOUR (0x02)</li> <li>• Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	number	Denotes the width of the runaround gap, if the object is a runaround object.
TailArrow	number	Set to 1 if the arrowhead is enabled at the beginning of the line. Set to 0, otherwise.
TintPercent	number	Specifies the tint percentage.
Unique	number	Denotes the internal identifier of the graphic object.
Width	number	Specifies the width of the object in points. The permissible range is 0.125 point to 3600 point.

## Arc methods

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

## AttrCondExpr

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

## AttrCondExpr methods

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

## Attribute

Property name	Data Type	Description
name	string	The name of the attribute.

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

## Attribute methods

[Attribute.](#)

## AttributeDef

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

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

## AttributeDef methods

[AttributeDef](#).

## AttributeDefs

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

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

## AttributeDefs methods

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

## AttributeEx

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

## Attributes

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

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

### Attributes methods

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

## AttributesEx

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

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

### AttributesEx methods

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

## BodyPage

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

### BodyPage methods

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

# Book

Property name	Data Type	Description
BookDontUpdateReferences	number	Set to 0 if FrameMaker is set to update cross-references when it opens the book; Set to 1, otherwise.
BooksModified	number	Set to 1 if the book has been modified; Set to 0, otherwise.
BooksSelected	number	Set to 1 if the book icon in the book window is selected; Set to 0 otherwise.
FirstComponentInBook	BookComponent	Denotes the first component in the book.
FirstSelectedComponentInBook	BookComponent	Denotes the first selected component in the book.
IsIconified	number	Set to 1 if the book window is iconified; set to 0 otherwise.
IsInFront	number	Set to 1 if the book window is in front of other windows in the FrameMaker product session; Set to 0 otherwise.
IsOnScreen	number	Set to 1 if the document is visible on the screen; Set to 0 otherwise.  Note: The IsOnScreen property is always 1 for books, and setting it to 0 has no effect.
Label	string	Denotes the title in the book window title bar.
Name	string	Indicates the path of the book.
NextOpenBookInSession	Book	Denotes the next open book in FrameMaker session's list of open books.
StatusLine	string	String that appears in the book status bar. Querying this string returns the status string.
ScreenHeight	number	Denotes the height of the book window in pixels.
ScreenWidth	number	Denotes the width of the book window in pixels.
ScreenX	number	Specifies the offset of the book window in pixels from the left side of the screen (or the left of the FrameMaker product application window).  If you set a value that results in the book window being off the screen, that value is ignored and the old value is retained.
ScreenY	number	Indicates the offset of the book window in pixels from the top of the screen (or the top of the FrameMaker product application window ).  If you set a value that results in the book window being off the screen, that value is ignored and the old value is retained.
TypeOfDisplayText	number	Specifies the type of text snippet to display for each icon in the book window. The two possible values are: <ul style="list-style-type: none"> <li>Constants.FV_BK_FILENAME (1)- displays the book component's filename</li> <li>Constants.FV_BK_TEXT (2)- displays the first paragraph of the component's first flow.</li> </ul>

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

Property name	Data Type	Description
PDFBookmarksOpenLevel	number	Denotes the level of bookmarks to be expanded when Acrobat opens the generated PDF document. The value can be any integer, or one of the following constants: <ul style="list-style-type: none"> <li>Constants.FV_PDFBookmarksOpenDefaultLevel (-1)</li> <li>Constants.FV_PDFBookmarksOpenAllLevels (-2)</li> <li>Constants.FV_PDFBookmarksOpenNoneLevel (-3)</li> </ul> If you specify an integer that is greater than the number of levels in the Bookmarks Settings, the Constants.PDFBookmarksOpenAllLevels (-2) value takes effect.
PDFConvertCMYKtoRGB	number	When set to 1, the PDFConvertCMYKtoRGB property corresponds with the setting "Convert CMYK colors to RGB" in the Save As PDF dialog box.
PDFDestsMarked	number	Set to 1 if the document has paragraphs or elements marked through the MarkedForNamedDestination property.  One of two things must occur in order for this property to be True: <ol style="list-style-type: none"> <li>The document must have been created and opened in FrameMaker 6.0 or later.</li> <li>The PDF FileSize Optimization client must have been run over it to mark all paragraphs or elements that are targets of hypertext links.</li> </ol>
PDFDistillerAbsent	number	A value of 1 indicates that there is no Acrobat Distiller available. Any other value indicates that Acrobat Distiller is available.
PDFDocInfo	Strings	Contains a list of strings expressing values to be set in the PDF Document Info dictionary when you save the book as PDF. Each dictionary entry is expressed as a pair of strings; the first string expresses the field name, and the second string expresses the field value.
PDFEndPage	string	Denotes the last page in the range of pages to be printed, in the FrameMaker numbering style.
PDFJobOption	string	Specifies the name of the Distiller Job Options. If the specified name does not exist in the Distiller Job Options list, then the first Distiller Job Option in the list is used.
PDFJobOptionsAbsent	number	A value of 1 indicates that PDF Job Options are not available. Any other value indicates that PDF Job Options are available.
PDFOpenPage	string	Denotes the PDF page number, in the FrameMaker numbering style, at which Acrobat opens the generated PDF document.
PDFPageHeight	number	Denotes the page height for the generated PDF document.
PDFPageWidth	number	Indicates the page width for the generated PDF document.



Property name	Data Type	Description
PDFPrintPageRange	number	Specifies the page range for generating the PDF document. If set, the PDF file contains only the specified pages. Else, FrameMaker generates PDF for the entire document or book.
PDFRegistrationMarks	number	Contains the registration marks for the generated PDF. The permissible values are: <ul style="list-style-type: none"> <li>• Constants.FV_PDFRegistrationMarksNone (0)</li> <li>• Constants.FV_PDFRegistrationMarksWestern (1)</li> <li>• Constants.FV_PDFRegistrationMarksTombo (2)</li> </ul>
PDFSeparateFiles	number	Set to 1, if a separate PDF file should be generated for each document in a book; Set to 0 otherwise. This property can be set for single documents, but is ignored in that case.
PDFStartPage	string	Denotes the first page of the printing page range, in the FrameMaker numbering style.
PDFZoomFactor	number	When the PDFZoomType property is set to Constants.FV_PDFZoomNone (0), the PDFZoomFactor property denotes the zoom percentage of the PDF document (metric 25% to 1600%). If the value is negative or zero, the value of Constants.FV_PDFZoomDefault (1) takes effect.
PDFZoomType	number	Denotes the PDF zoom setting at which Acrobat opens the generated PDF document. Permissible values are: <ul style="list-style-type: none"> <li>• Constants.FV_PDFZoomNone (0)</li> <li>• Constants.FV_PDFZoomDefault (1)</li> <li>• Constants.FV_PDFZoomPage (2)</li> <li>• Constants.FV_PDFZoomWidth (3)</li> <li>• Constants.FV_PDFZoomHeight (4)</li> </ul> If a value other than these is specified, Constants.FV_PDFZoomDefault (1) takes effect.
PDFAllNamedDestinations	number	Set to 1 if the generated PDF document should include Named Destinations for every paragraph and FrameMaker structure element in the book. This results in a larger PDF filesize.  If set to 0, the generated PDF document will have Named Destinations only for those paragraphs and objects that have already been marked with PDFDestsMarked set to True.
PrintBlankPages	number	Set to 1 if the PageRounding property is set to allow empty pages at the end of documents; set to 0 otherwise.
PrintCollated	number	Set to 1 if Collate is enabled; Set to 0 otherwise.
PrintEmulsion	number	Denotes the direction of the print emulsion. The possible value are: <ul style="list-style-type: none"> <li>• Constants.FV_EMUL_UP (0) - Emulsion side up</li> <li>• Constants.FV_EMUL_DOWN (1) - Emulsion side down</li> </ul>

Property name	Data Type	Description
PrinterName	string	This property has no effect.
PrintEvenPages	number	Set to 1 if Print Even-Numbered Pages is enabled.
PrintFileName	string	Specifies the filename to print to. You can set the filename to be the default filename by specifying the <code>PrintFileName</code> value as null.
PrintImaging	number	Denotes the type of print imaging. The permissible values are: <ul style="list-style-type: none"> <li>• Constants.FV_IMG_POSITIVE (0)</li> <li>• Constants.FV_IMG_NEGATIVE (1)</li> </ul>
PrintLastSheetFirst	number	Set to 1 to print the last sheet first. Else, set to 0.
PrintLowRes	number	Set to 1 to print in low resolution. Else, set to 0.
PrintNumCopies	number	Denotes the number of copies to print.
PrintOddPages	number	Set to 1 if Print Odd-Numbered Pages is enabled.
PrintPaperHeight	number	Specifies the height of the printing paper.
PrintPaperWidth	number	Indicates the width of the printing paper.
PrintRegistrationMarks	number	Set to 1 if Registration Marks is enabled; Set to 0 otherwise.
PrintScale	number	Denotes the Print scale factor expressed as a percentage metric (0% to 100%).  For print scale percentages, the value <code>1&lt;&lt;16 or 0x10000</code> specifies 100%.
PrintSeps	number	Set to 1 if Print Separations is enabled.
PrintToFile	number	Set to 1 if Print Only to File is enabled.
SkipBlankSeps	number	Set to 1 if Skip Blank Separations (do not print blank color separations) is enabled.
CustomElementList	Strings	Contains the list of tags to display when <code>ElementCatalogDisplay</code> is set to <code>Constants.FV_ELCAT_CUSTOM (0x04)</code> .
ElementCatalog	ElementCatalogEntries	Contains a list of elements in the Element Catalog.
ElementCatalogDisplay	number	Catalog display options. Show tags for: <ul style="list-style-type: none"> <li>• Constants.FV_ELCAT_STRICT (0x00): Valid children for working from start to finish.</li> <li>• Constants.FV_ELCAT_LOOSE (0x01): Valid children for working in any order.</li> <li>• Constants.FV_ELCAT_CHILDREN (0x02): Children allowed anywhere in parent.</li> <li>• Constants.FV_ELCAT_ALL (0x03): All elements.</li> <li>• Constants.FV_ELCAT_CUSTOM (0x04): List of tags specified by the <code>CustomElementList</code> property.</li> </ul>

Property name	Data Type	Description
FileExtensionOverride	string	Denotes the filename extension to use when saving the document as XML. Typically, this property is used to save XHTML with a .htm extension rather than .xml. This setting should be set in the structure application for the document's DOCTYPE.
ElementSelection	ElementRange	Indicates the currently selected element range in the book.
FirstFmtChangeListInDoc	FmtChangeList	Specifies the ID of the first format change list in the list of format change lists in the book.
FirstElementDefInDoc	ElementDef	Denotes the first element definition in the list of element definitions in the document (FO_ElementDef ID).
HighestLevelElement	Element	Denotes the top most element for a structured book/document.
NewElemAttrDisplay	number	Specifies the attribute display properties for new elements. The permissible values are: <ul style="list-style-type: none"> <li>Constants.FV_ATTR_DISP_NONE (1): Do not display attributes.</li> <li>Constants.FV_ATTR_DISP_REQSPEC (2): Display required and specified attributes.</li> <li>Constants.FV_ATTR_DISP_ALL (3): Display all attributes.</li> </ul>
NewElemAttrEditing	number	Specifies when the Edit Attributes dialog box appears for new elements. The permissible values are: <ul style="list-style-type: none"> <li>Constants.FV_ATTR_EDIT_NONE (0)</li> <li>Constants.FV_ATTR_EDIT_REQUIRED (1)</li> <li>Constants.FV_ATTR_EDIT_ALWAYS (2)</li> </ul>
SeparateInclusions	number	Set to 1 if inclusions are listed separately in the element catalog; set to 0 otherwise.
StructuredApplication	string	Contains the name of the structured application that is associated with the book. If the book has no associated structure application, GetString() returns an empty string.
UseInitialStructure	number	Set to 1 if FrameMaker inserts the initial structure for new elements; set to 0 otherwise.
XmlDocType	string	Contains the DOCTYPE parameter from the source XML.
XmlEncoding	string	Specifies the encoding parameter of the XML Declaration for the source XML. The string is empty if no encoding is specified. If this property is set, the XML Declaration contains the encoding parameter with this value on Save As XML.
XmlFileEncoding	string	Specifies the encoding that is detected for the source XML book. If no encoding is specified for the source XML, XmlEncoding is an empty string. In that case, if this value is set, it determines the encoding to use when saving as XML. If XmlEncoding has a value, this string may be empty.

Property name	Data Type	Description
XmlPublicId	string	Contains the DOCTYPE public identifier for the source XML document.
XmlStandAlone	number	<p>Indicates an integer that specifies the XML standalone parameter for the XML document, that is the source of the current document. The permissible values are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_XML_STANDALONE_YES (1)</li> <li>• Constants.FV_XML_STANDALONE_NO (2)</li> <li>• Constants.FV_XML_STANDALONE_NONE (3)</li> <li>• Constants.FV_XML_STANDALONE_NODEC (4)</li> </ul> <p>The standalone parameter is declared in the XML Declaration. For a file with no XML declaration, the value is Constants.FV_XML_STANDALONE_NODEC (4). For an XML declaration with no standalone parameter, this value is Constants.FV_XML_STANDALONE_NONE (3).</p>
XmlStyleSheet	string	<p>Specifies the XML stylesheet processing instruction to write out to XML when saving the book as XML.</p> <p>Note; The ESTK does not verify that you are using the correct syntax in this string.</p> <p>The string you set should not include the PI delimiters, &lt;? and ?&gt;. For example, the string you supply for my.css may be:</p> <pre>"type=\"text\\css\" href=\"my.css\""</pre> <p>Only use this string to set a specific stylesheet specification. GetString() always returns null for this parameter. To get the list of stylesheet specifications associated with a book, use XmlStyleSheetList.</p>
XmlStyleSheetList	Strings	<p>Contains a list of stylesheet processing instructions for the current book. One book can have more than one stylesheet specification associated with it.</p> <p>Note: The ESTK does not verify that you are using the correct syntax in these strings.</p> <p>The strings should not include the PI delimiters, &lt;? and ?&gt;.</p> <p>For example, the string you supply for my.css may be:</p> <pre>"type=\"text\\css\" href=\"my.css\""</pre> <p>Setting a list as the value to the XmlStyleSheetList property completely overwrites the preceding list.</p>
XmlSystemId	string	Contains the DOCTYPE system identifier for the source XML document.

Property name	Data Type	Description
XmlUseBOM	number	<p>Indicates whether a byte order mark was detected when opening the source XML. The permissible values are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_XML_USEBOM_YES (1)</li> <li>• Constants.FV_XML_USEBOM_NO (2)</li> <li>• Constants.FV_XML_USEBOM_UTF8 (3)</li> <li>• Constants.FV_XML_USEBOM_UTF16BE (4)</li> <li>• Constants.FV_XML_USEBOM_UTF16LE (5)</li> <li>• Constants.FV_XML_USEBOM_UTF32BE (6)</li> <li>• Constants.FV_XML_USEBOM_UTF32LE (7)</li> </ul> <p>When saving as XML, if this property is set to Constants.FV_XML_USEBOM_YES (1), FrameMaker writes a byte order mark in the resulting XML.</p>
XmlVersion	string	<p>Depicts the XML Version that is specified in the XML Declaration when the file is opened. If no XML version is specified, <code>GetString()</code> returns an empty string.</p> <p>If this string contains an invalid XML declaration, a parsing error occurs when the book is saved as XML.</p>
XmlWellFormed	number	<p>Indicates whether the source XML is qualified as well formed. The permissible values are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_XML_WELLFORMED_YES (1)</li> <li>• Constants.FV_XML_WELLFORMED_NO (2)</li> </ul>
BooksViewOnly	number	Set to 1 if the book is view-only; set to 0 otherwise.
ViewOnlyDeadCodes	<code>UInts</code>	Contains a list of F-codes that cannot be executed in the book.
ViewOnlyWinBorders	number	Set to 1 if the book has normal window borders; Set to 0 if the book's border buttons are suppressed.
ViewOnlyWinPopup	number	Set to 1 if the book window pop-up menu is available; set to 0 otherwise.

Property name	Data Type	Description
PDFGenerateForReview	number	This property is used to generate a PDF for review workflow that is getting review comments in PDF and importing them back to FrameMaker document.  The property helps in importing back the comments correctly. When this flag is turned ON, the behavior is same as if the "Generate for review only" checkbox is ON in the PDF setup dialog.
ShowElementDescriptiveNames	bool	Set to True to show descriptive names for elements; Set to False, otherwise.  <i>Note: The ShowElementDescriptiveNames property is applicable only for structured FrameMaker.</i>
UseInitialStructureOfAutoInsertedElements	bool	If set to True, auto-insertion rules will be processed recursively. For example: If an element is inserted automatically, and auto-insertion rules exist for this element, in the element definition, then those rules would also be processed.

## Book methods

[Close](#), [Compare](#), [GetProps](#), [GetUniqueObject](#), [Import](#), [NewBookComponentInHierarchy](#), [NewBookComponentOfTypeInHierarchy](#), [NewSeriesBookComponent](#), [NewSeriesObject](#), [ObjectValid](#), [Save](#), [SetProps](#), [SilentPrintDoc](#), [SimpleGenerate](#), [SimpleImportElementDefs](#), [SimpleImportFormats](#), [SimpleSave](#), [UpdateBook](#).

# BookComponent

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

Property name	Data Type	Description
ChapNumComputeMethod	number	<p>The component document's chapter numbering type. The possible values are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_NUM_READ_FROM_FILE (0x00) - Use the numbering properties from the document associated with this book component.</li> <li>• Constants.FV_NUM_CONTINUE (0x01) - Continue numbering from the previous chapter.</li> <li>• Constants.FV_NUM_RESTART (0x02) - Use the value specified for the ChapterNumber property.</li> <li>• Constants.FV_NUM_SAME (0x03) - Use the same chapter number as for the previous file.</li> </ul>
ChapterNumber	number	<p>If the ChapNumComputeMethod property is set to Constants.FV_NUM_RESTART (0x02), use the ChapterNumber property as the chapter number.</p>
ChapterNumStyle	number	<p>Denotes the chapter numbering style. The possible value are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_NUMSTYLE_NUMERIC (0x00) - Arabic</li> <li>• Constants.FV_NUMSTYLE_ROMAN_UC (0x01) - Roman, uppercase</li> <li>• Constants.FV_NUMSTYLE_ROMAN_LC (0x02) - Roman, lowercase</li> <li>• Constants.FV_NUMSTYLE_ALPHA_UC (0x03) - Alphabetic, uppercase</li> <li>• Constants.FV_NUMSTYLE_ALPHA_LC (0x04) - Alphabetic, lowercase</li> <li>• Constants.FV_NUMSTYLE_KANJI (0x05) - Kanji</li> <li>• Constants.FV_NUMSTYLE_ZENKAKU (0x06) - Zenkaku</li> <li>• Constants.FV_NUMSTYLE_ZENKAKU_UC (0x07) - Zenkaku, uppercase</li> <li>• Constants.FV_NUMSTYLE_ZENKAKU_LC (0x08) - Zenkaku, lowercase</li> <li>• Constants.FV_NUMSTYLE_KANJI_KAZU (0x09) - Kazu</li> <li>• Constants.FV_NUMSTYLE_DAIJI (0x0a) - Daiji</li> <li>• Constants.FV_NUMSTYLE_TEXT (0x0b) - Text. This property is only for volume/chapter numbers.</li> </ul>
ChapterNumText	string	<p>If the ChapNumStyle property is set to Constants.FV_NUMSTYLE_TEXT (0x0b), use the string specified in the ChapterNumText property as the chapter number.</p>



Property name	Data Type	Description
ComponentDisplayText	string	Specifies the text that displays in the book window when the value of the TypeOfDisplayText property is set to Constants.FV_BK_TEXT (2).  Set the ComponentDisplayText property to an empty string (" ") to allow FrameMaker to automatically update the text line with normal information.
ComponentIsSelected	number	Set to 1 if the component is selected in the book window; otherwise set to 0.
ExtractTags	Strings	List of paragraph tags or markers type names that are used to set up a generatable file (for example, table of contents, list of figures, standard index or index of authors).
FirstPageNum	number	Number for the first page in the component; used when the PageNumComputeMethod property is set to Constants.FV_NUM_RESTART (0x02).
FnFirstNum	string	Number for the first footnote in the component; used when the FnNumComputeMethod property is set to Constants.FV_NUM_RESTART (0x02).
FnCustNumString	string	Indicates the characters for custom document footnote numbers.
FnNumStyle	number	Footnote numbering style. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_FN_NUM_NUMERIC (0x00) - Arabic</li> <li>• Constants.FV_FN_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>• Constants.FV_FN_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>• Constants.FV_FN_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_FN_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>• Constants.FV_FN_NUM_CUSTOM (0x05) - Custom numbering</li> <li>• Constants.FV_FN_NUM_KANJI (0x06) - Kanji characters</li> <li>• Constants.FV_FN_NUM_ZENKAKU (0x07) - Zenkaku</li> <li>• Constants.FV_FN_NUM_ZENKAKU_UC (0x08) - Zenkaku uppercase</li> <li>• Constants.FV_FN_NUM_ZENKAKU_LC (0x09) - Zenkaku lowercase</li> <li>• Constants.FV_FN_NUM_KANJI_KAZU (0x0A) - Kazu</li> <li>• Constants.FV_FN_NUM_DAIJI (0x0B) - Daiji</li> </ul>

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

Property name	Data Type	Description
PageNumStyle	number	<p>Page numbering style. The possible values are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_PAGE_NUM_NUMERIC (0x00) - Arabic</li> <li>• Constants.FV_PAGE_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>• Constants.FV_PAGE_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>• Constants.FV_PAGE_NUM_KANJI (0x05) - Kanji characters</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU (0x06) - Zenkaku</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU_UC (0x07) - Zenkaku uppercase</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU_LC (0x08) - Zenkaku lowercase</li> <li>• Constants.FV_PAGE_NUM_KANJI_KAZU (0x09) - Kazu</li> <li>• Constants.FV_PAGE_NUM_DAIJI (0x0a) - Daiji</li> </ul>
PageSide	number	<p>Specifies the page side on which to start the component document. The possible values are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_BK_START_FROM_FILE (0x01)</li> <li>• Constants.FV_BK_START_NEXT_AVAILABLE (0x02)</li> <li>• Constants.FV_BK_START_LEFT (0x03)</li> <li>• Constants.FV_BK_START_RIGHT (0x04)</li> </ul>
PgfNumComputeMethod	number	<p>The type of numbering for the component document's paragraph. Possible values are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_NUM_READ_FROM_FILE (0x00) - Use the numbering properties from the document associated with this book component.</li> <li>• Constants.FV_NUM_CONTINUE (0x01) - Continue the numbering from the previous file.</li> <li>• Constants.FV_NUM_RESTART (0x02) - Restart numbering at 1.</li> </ul>
PrevComponentInBook	<a href="#">BookComponent</a>	Denotes the previous component in the book file.
PrintInclude	number	Set to 1 if the component document is included in the list of book files to be printed; otherwise set to 0.
TblFnCustNumString	string	Specifies the characters for custom table footnote numbers.

Property name	Data Type	Description
TblFnNumStyle	number	Table footnote numbering style. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_FN_NUM_NUMERIC (0x00) - Arabic</li> <li>• Constants.FV_FN_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>• Constants.FV_FN_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>• Constants.FV_FN_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_FN_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>• Constants.FV_FN_NUM_CUSTOM (0x05)- Custom numbering</li> <li>• Constants.FV_FN_NUM_KANJI (0x06) - Kanji characters</li> <li>• Constants.FV_FN_NUM_ZENKAKU (0x07) - Zenkaku</li> <li>• Constants.FV_FN_NUM_ZENKAKU_UC (0x08) - Zenkaku uppercase</li> <li>• Constants.FV_FN_NUM_ZENKAKU_LC (0x09) - Zenkaku lowercase</li> <li>• Constants.FV_FN_NUM_KANJI_KAZU (0x0A) - Kazu</li> <li>• Constants.FV_FN_NUM_DAIJI (0x0B) - Daiji</li> </ul>
TblFnNumComputeMethod	number	The numbering type of the component document's table footnote. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_NUM_READ_FROM_FILE (0x00) - Use the numbering properties from the document associated with this book component.</li> <li>• Constants.FV_NUM_RESTART (0x02) - Start at 1.</li> </ul>
Unique	number	Represents the Unique ID of the book component.
VolNumComputeMethod	number	The component document's volume numbering type. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_NUM_READ_FROM_FILE (0x00) - Use the numbering properties from the document associated with this book component.</li> <li>• Constants.FV_NUM_CONTINUE (0x01) - Continue numbering from the previous volume.</li> <li>• Constants.FV_NUM_RESTART (0x02) - Use the value specified for the VolumeNumber property.</li> <li>• Constants.FV_NUM_SAME (0x03) - Use the same volume number as for the previous file.</li> </ul>
VolumeNumber	number	If the VolNumComputeMethod property is set to Constants.FV_NUM_RESTART (0x02), use the value specified in the VolumeNumber property as the volume number.

Property name	Data Type	Description
VolumeNumStyle	number	The volume numbering style. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_NUMSTYLE_NUMERIC (0x00) - Arabic</li> <li>• Constants.FV_NUMSTYLE_ROMAN_UC (0x01) - Roman uppercase</li> <li>• Constants.FV_NUMSTYLE_ROMAN_LC (0x02) - Roman lowercase</li> <li>• Constants.FV_NUMSTYLE_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_NUMSTYLE_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>• Constants.FV_NUMSTYLE_KANJI (0x05) - Kanji</li> <li>• Constants.FV_NUMSTYLE_ZENKAKU (0x06) - Zenkaku</li> <li>• Constants.FV_NUMSTYLE_ZENKAKU_UC (0x07) - Zenkaku uppercase</li> <li>• Constants.FV_NUMSTYLE_ZENKAKU_LC (0x08) - Zenkaku lowercase</li> <li>• Constants.FV_NUMSTYLE_KANJI_KAZU (0x09) - Kazu</li> <li>• Constants.FV_NUMSTYLE_DAIJI (0x0a) - Daiji</li> <li>• Constants.FV_NUMSTYLE_TEXT (0x0b) - Text</li> </ul>
VolumeNumText	string	If the <code>VolNumStyle</code> property is set to <code>Constants.FV_NUMSTYLE_TEXT (0x0b)</code> , use the string specified in the <code>VolumeNumText</code> property as the chapter number.
ComponentElement	<a href="#">Element</a>	Specifies the component element (FO_Element ID).
ExtractElementTags	<a href="#">Strings</a>	List of element tags that are used to set up a generatable file (for example, table of contents, list of figures, or list of tables).
FirstComponentInBookComponent	<a href="#">BookComponent</a>	Denotes the first component of the hierarchy for which the current component is the parent.
BookComponentParent	<a href="#">BookComponent</a>	Denotes the parent of the current book element.
ExcludeBookComponent	number	Specifies the component to exclude when printing the book.
BookComponentTemplatePath	string	The path to the file that is used as a template for a folder.
BookComponentTitle	string	The name of a folder or a group component.

Property name	Data Type	Description
ComponentType	number	Indicates whether the component is a folder, group, or a document. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_BK_FOLDER (0x02) - Folder</li> <li>• Constants.FV_BK_GROUP (0x40) - Group</li> <li>• Constants.FV_BK_FM (0x8) - FrameMaker document</li> <li>• Constants.FV_BK_MIF (0x10) - MIF document</li> <li>• Constants.FV_BK_XML (0x20) - XML document</li> </ul>
XmlApplicationForBookComponent	string	The application name for the XML that is set as template for the folder.
SecNumComputeMethod	number	Indicates the type of numbering applied to a particular book component. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_NUM_CONTINUE (0x01)</li> <li>• Constants.FV_NUM_RESTART (0x02)</li> <li>• Constants.FV_NUM_SAME (0x03)</li> </ul>
SectionNumStyle	number	Denotes the section numbering style. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_PAGE_NUM_NUMERIC (0x00) - Arabic</li> <li>• Constants.FV_PAGE_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>• Constants.FV_PAGE_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>• Constants.FV_PAGE_NUM_KANJI (0x05) - Kanji characters</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU (0x06) - Zenkaku</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU_UC (0x07) - Zenkaku uppercase</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU_LC (0x08) - Zenkaku lowercase</li> <li>• Constants.FV_PAGE_NUM_KANJI_KAZU (0x09) - Kazu</li> <li>• Constants.FV_PAGE_NUM_DAIJI (0x0a) - Daiji</li> </ul>
SectionNumber	number	Denotes the section number of the specified component.
SectionNumText	string	The text to use for the SectionNumber property.

Property name	Data Type	Description
SubsecNumComputeMethod	number	Indicates the type of numbering applied to a particular book component. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_NUM_CONTINUE (0x01)</li> <li>• Constants.FV_NUM_RESTART (0x02)</li> <li>• Constants.FV_NUM_SAME (0x03)</li> </ul>
SubsectionNumStyle	number	Denotes the subsection numbering style. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_PAGE_NUM_NUMERIC (0x00) - Arabic</li> <li>• Constants.FV_PAGE_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>• Constants.FV_PAGE_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>• Constants.FV_PAGE_NUM_KANJI (0x05) - Kanji characters</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU (0x06) - Zenkaku</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU_UC (0x07) - Zenkaku uppercase</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU_LC (0x08) - Zenkaku lowercase</li> <li>• Constants.FV_PAGE_NUM_KANJI_KAZU (0x09) - Kazu</li> <li>• Constants.FV_PAGE_NUM_DAIJI (0x0a) - Daiji</li> </ul>
SubsectionNumber	number	Denotes the subsection number of the specified component.
SubsectionNumText	string	The text to use for the <code>SubsectionNumber</code> property.
NextBookComponentInDFSOrder	<a href="#">BookComponent</a>	Denotes the pointer to traverse through book components in a hierarchal manner (top to bottom).
BookComponentFileType	number	Determines the file type of the document. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_BK_FM (0x8) - FrameMaker document</li> <li>• Constants.FV_BK_MIF (0x10) - MIF document</li> <li>• Constants.FV_BK_XML (0x20) - XML document</li> <li>• Constants.FV_BK_XML (0x20) - Ditamap</li> </ul>
PrevBookComponentInDFSOrder	<a href="#">BookComponent</a>	Denotes the pointer to traverse through book components in an heirarchial manner (bottom to top).

Property name	Data Type	Description
BookComponentIsFolderWithTemplate	number	Determines whether a template is associated with a book. Returns 1 if the <code>BookComponent</code> is a folder and a template is associated with the component. Returns 0, otherwise.
BookComponentIsFolderWithoutTemplate	number	Determines whether a template is not associated with a book. Returns 1 if the <code>BookComponent</code> is a folder and a template is not associated. Returns 0 otherwise.
ComponentIsDitaMap	bool	Determines whether a book component is a ditamap. Set to 1 if the book component is a ditamap; set to 0 otherwise.

## BookComponent methods

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

## Cell

Property name	Data Type	Description
CellAboveInCol	<a href="#">Cell</a>	Denotes the cell above the current cell (FO_CellID).
CellAngle	number	Indicates the angle of rotation of the cell.
CellBelowInCol	<a href="#">Cell</a>	Denotes the cell below the current cell (FO_CellID).
CellColNum	number	Represents the cell's column number.
CellDefaultBottomRuling	<a href="#">RulingFmt</a>	Indicates the cell's default bottom ruling (FO_RulingFmt ID).
CellDefaultLeftRuling	<a href="#">RulingFmt</a>	Specifies the cell's default left ruling (FO_RulingFmt ID).
CellDefaultRightRuling	<a href="#">RulingFmt</a>	Indicates the cell's default right ruling (FO_RulingFmt ID).
CellDefaultTopRuling	<a href="#">RulingFmt</a>	Represents the cell's default top ruling (FO_RulingFmt ID).
CellsShown	number	Returns 1 if the cell is conditional and is visible; returns 0 otherwise.
CellsStraddled	number	Returns 1 if the cell is in a straddle but is not the first cell; returns 0 if the cell is the first cell in a straddle, or is not in a straddle.
CellNumColsStraddled	number	Specifies the number of columns in the straddle, if the cell is the first cell in a horizontal straddle; Otherwise the value is 1.
CellNumRowsStraddled	number	Specifies the number of rows in the straddle, if the cell is the first cell in a vertical straddle; Otherwise the value is 1.
ContentHeight	number	Denotes the distance between the top of the cell and the baseline of the last line in the cell.
CellOverrideBottomRuling	<a href="#">RulingFmt</a>	Represents the cell's bottom ruling (FO_RulingFmt ID), if there is an override. The value is null if there is no override.



Property name	Data Type	Description
CellOverrideFill	number	Indicates the cell's fill pattern. The value is null if there is no override fill pattern.
CellOverrideLeftRuling	RulingFmt	Indicates the cell's left ruling (FO_RulingFmt ID), if there is an override. The value is null if there is no override.
CellOverrideRightRuling	RulingFmt	Specifies the cell's right ruling (FO_RulingFmt ID), if there is an override. The value is null if there is no override.
CellOverrideShading	Color	Denotes the cell's spot color (FO_Color ID). The value is null if there is no override shading.
CellOverrideTopRuling	RulingFmt	Denotes the cell's top ruling (FO_RulingFmt ID), if there is an override. The value is null if there is no override.
CellRow	Row	Denotes the row containing the cell (FO_RowID).
CellUseOverrideBRuling	number	Set to 1 if the cell's bottom ruling (specified by the CellOverrideBottomRuling property) overrides the default ruling specified by the table format; set to 0 otherwise.
CellUseOverrideFill	number	Set to 1 if the cell's fill pattern (specified by the CellOverrideFill property) overrides the default fill pattern specified by the table format; set to 0 otherwise.
CellUseOverrideLRuling	number	Set to 1 if the cell's left ruling (specified by the CellOverrideLeftRuling property) overrides the ruling specified by the table format; set to 0 otherwise.
CellUseOverrideRRuling	number	Set to 1 if the cell's right ruling (specified by the CellOverrideRightRuling property) overrides the ruling specified by the table format; set to 0 otherwise.
CellUseOverrideShading	number	Set to 1 if the cell's shading (specified by the CellOverrideShading property) overrides the default shading specified by the table format; set to 0 otherwise.
CellUseOverrideTRuling	number	Set to 1 if the cell's top ruling (specified by the CellOverrideTopRuling property) overrides the default top ruling specified by the table format; set to 0 otherwise.
Element	Element	Denotes the element containing the cell, if the cell is in a FrameMaker document.
FirstPgf	Pgf	Represents the first paragraph in the cell (FO_Pgf ID).
InTextFrame	TextFrame	Represents the text frame containing the cell (FO_TextFrame ID).
InTextObj	SubCol	Represents the text object containing the cell (FO_SubCol ID).
LastPgf	Pgf	Indicates the last paragraph in the cell (FO_Pgf ID).
NextCellInRow	Cell	Denotes the next cell in current row from left to right (FO_Cell ID).
NextCellInTbl	Cell	Represents the next cell from left to right (FO_Cell ID). If the cell is at the end of a row, the next cell is the first cell in the next row.
NextCell	Cell	Represents the next cell in the text frame (FO_Cell ID).

Property name	Data Type	Description
Overflowed	number	Specifies whether the text in the cell overflows. Set to 1 if the row Height Limit Maximum is too low to display all the text in the cell; set to 0 otherwise.
PrevCellInRow	Cell	Represents the previous cell in the current row (FO_Cell ID).
PrevCell	Cell	Denotes the previous cell in the text frame (FO_Cell ID).
Unique	number	Represents the cell's unique ID.

## Cell methods

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

## CharFmt

Property name	Data Type	Description
Capitalization	number	The capitalization type. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CAPITAL_CASE_NORM (0) - Normal capitalization (mixed uppercase and lowercase)</li> <li>• Constants.FV_CAPITAL_CASE_SMALL (1) - Small caps</li> <li>• Constants.FV_CAPITAL_CASE_LOWER (2) - Lowercase letters only</li> <li>• Constants.FV_CAPITAL_CASE_UPPER (3) - Uppercase letters only</li> </ul>
ChangeBar	number	Returns 1 if Change Bars are on.
CharTag	string	Denotes the character format's tag name.
Color	Color	Indicates the spot color ((FO_Color ID).
CombinedFontFamily	CombinedFontDefn	Denotes the combined font definition (FO_CombinedFontDefn).
FontAngle	number	Represents the font angle (specifies an index into the array of font angles provided by the session property FontAngleNames).
FontEncodingName	string	Indicates the encoding of the font.
FontFamily	number	Specifies the font family (specifies an index into the array of font families provided by the session property FontFamilyNames).
FontPlatformName	string	Name that uniquely identifies a font on a specific platform. For combined fonts, the name is the Asian font name.
FontPostScriptName	string	Name given to a font when it is sent to a PostScript printer. For combined fonts, the name is the Asian font name.

Property name	Data Type	Description
WesternFontPlatformName	string	Name that uniquely identifies the Roman component of a combined font on a specific platform.
WesternFontPostScriptName	string	Name given to the Roman component of a combined font when it is sent to a PostScript printer.
FontSize	number	Denotes the font size (2 point to 400 point).
FontVariation	number	Indicates the font variation (specifies an index into the array of font variations provided by the session property <code>FontVariationNames</code> ).
FontWeight	number	Indicates the font weight (specifies an index into the array of font weights provided by the session property <code>FontWeightNames</code> ).
Language	number	Denotes the hyphenation and spell-checking language to use. The possible languages are: <ul style="list-style-type: none"> <li>• Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li>• Constants.FV_LANG_ENGLISH (0x01)</li> <li>• Constants.FV_LANG_BRITISH (0x02)</li> <li>• Constants.FV_LANG_GERMAN (0x03)</li> <li>• Constants.FV_LANG_SWISS_GERMAN (0x04)</li> <li>• Constants.FV_LANG_FRENCH (0x05)</li> <li>• Constants.FV_LANG_CANADIAN_FRENCH (0x06)</li> <li>• Constants.FV_LANG_SPANISH (0x07)</li> <li>• Constants.FV_LANG_CATALAN (0x08)</li> <li>• Constants.FV_LANG_ITALIAN (0x09)</li> <li>• Constants.FV_LANG_FINNISH (0x10)</li> <li>• Constants.FV_LANG_SWEDISH (0x11)</li> <li>• Constants.FV_LANG_JAPANESE (0x12)</li> <li>• Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>• Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>• Constants.FV_LANG_KOREAN (0x15)</li> <li>• Constants.FV_LANG_PORTUGUESE (0x0A)</li> <li>• Constants.FV_LANG_BRAZILIAN (0x0B)</li> <li>• Constants.FV_LANG_DANISH (0x0C)</li> <li>• Constants.FV_LANG_DUTCH (0x0D)</li> <li>• Constants.FV_LANG_NORWEGIAN (0x0E)</li> <li>• Constants.FV_LANG_NYNORSK (0x0F)</li> </ul>

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

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

## CharFmt methods

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

## CMSCheckinParam

Property name	Data Type	Description
version	int	Denotes the version of the checked-in document. This property uses values from the <code>CMSVersioningStrategy</code> enumeration:  <code>CMSSameVersion</code> : Same version <code>CMSMinorVersion</code> : Minor version <code>CMSMajorVersion</code> : Major version
versionLabel	string	Specifies the label of the document version

Property name	Data Type	Description
description	string	Specifies the description of the document version
checkinComment	string	Specifies the check-in comments of the document version
makeThisCurrentVersion	bool	Determines whether the specified version is the current version. Set to True to make this checked-in document the current version; else set to False

## CMSDeleteParam

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

## CMSInfo

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

## CMSMenuItem

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

## CMSProperty

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

## CMSProperties

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

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

## CMSRegistration

Property name	Data Type	Description
Capitalization	number	The capitalization type. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CAPITAL_CASE_NORM (0)</li> <li>• Constants.FV_CAPITAL_CASE_SMALL (1)</li> <li>• Constants.FV_CAPITAL_CASE_LOWER (2)</li> <li>• Constants.FV_CAPITAL_CASE_UPPER (3)</li> </ul>

## CMSResult

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

## CMSSession

Property name	Data Type	Description
Capitalization	number	The capitalization type. The possible values are: <ul style="list-style-type: none"><li>• Constants.FV_CAPITAL_CASE_NORM (0)</li><li>• Constants.FV_CAPITAL_CASE_SMALL (1)</li><li>• Constants.FV_CAPITAL_CASE_LOWER (2)</li><li>• Constants.FV_CAPITAL_CASE_UPPER (3)</li></ul>



## Color

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

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

## Color methods

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

## CombinedFont

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

## CombinedFont methods

[CombinedFont](#).

## CombinedFontDefn

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

### CombinedFontDefn methods

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

## CombinedFonts

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

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

### CombinedFonts methods

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

# Command

Property name	Data Type	Description
CanHaveCheckMark	number	Returns True if the menu item can have a check mark. If the menu item is defined by FrameMaker, you can read this property, but not set it.
CheckMarksOn	number	Returns 1 if the menu item can have a check mark and the check mark is on. If the menu item is defined by FrameMaker, you can read this property, but not set it.
CommandNum	number	Represents the integer that you specified for the <code>cmd</code> parameter of <code>DefineAndAddCommand()</code> or <code>DefineCommand()</code> . When the user executes the command, FrameMaker passes this integer to your client's <code>Command()</code> function.  If the menu item is defined by FrameMaker, you can read this property, but not set it.
EnabledWhen	number	The context in which the menu item is enabled.. If the menu item is defined by FrameMaker, you can read this property, but not set it.  The following are the values that the <code>EnabledWhen</code> property can have and the corresponding contexts in which a menu item is active. <ul style="list-style-type: none"> <li>• Constants.FV_ENABLE_OBJ_PROPS (8) - The insertion point is in text, a table, or a math object, or a graphic object is selected.</li> <li>• Constants.FV_ENABLE_IN_TABLE (9) - The insertion point or selection is in any part of a table.</li> <li>• Constants.FV_ENABLE_IN_TABLE_TITLE (10) - The insertion point or selection is in the table title.</li> <li>• Constants.FV_ENABLE_IN_CELL_TEXT (11) - The insertion point or selection is in a table cell.</li> <li>• Constants.FV_ENABLE_IS_CELL (12) - A single cell in a table is selected.</li> <li>• Constants.FV_ENABLE_IS_CELLS (13) - One or more cells in a table are selected.</li> </ul>

Property name	Data Type	Description
		<ul style="list-style-type: none"> <li>• Constants.FV_ENABLE_IS_TABLE(14) - An entire table is selected.</li> <li>• Constants.FV_ENABLE_IS_OBJ (15) - An object is selected.</li> <li>• Constants.FV_ENABLE_IS_TEXT_FRAME (16) - A text frame is selected.</li> <li>• Constants.FV_ENABLE_IS_OR_IN_FRAME (17) - The selected object is a graphic frame or is in a graphic frame that is not a page frame.</li> <li>• Constants.FV_ENABLE_IS_AFRAME (18) - The first selected object is an anchored frame.</li> <li>• Constants.FV_ENABLE_IS_TEXT_INSET (19) - The first selected object is a text inset.</li> <li>• Constants.FV_ENABLE_IS_GRAPHIC_INSET (20) - The first selected object is a graphic inset.</li> <li>• Constants.FV_ENABLE_IN_FLOW (21) - A text frame is selected, or the insertion point or selection is in a paragraph.</li> <li>• Constants.FV_ENABLE_COPY (22) - Some text or an object is selected.</li> </ul>
		<ul style="list-style-type: none"> <li>• Constants.FV_ENABLE_COPY_FONT (23) - The insertion point or selection is in the text of a paragraph, a math object, a table, or a text line.</li> <li>• Constants.FV_ENABLE_CAN_PASTE(24) - The Clipboard contains an object or text that can be pasted at the insertion point.</li> <li>• Constants.FV_ENABLE_IS_VIEW_ONLY (25) - The current document is locked.</li> <li>• Constants.FV_ENABLE_NEEDS_DOCP_ONLY (26) - A document is open.</li> <li>• Constants.FV_ENABLE_NEEDS_BOOKP_ONLY (27) - A book is open.</li> <li>• Constants.FV_ENABLE_NEEDS_DOCP_OR_BOOKP (28) - A document or a book is open.</li> </ul>
ExpandOMaticParent	Command	Denotes the virtual parent object, if the menu item is an expandomatic menu item.
Fcode	uint	Specifies an f-code that FrameMaker executes when the user chooses the menu item or presses the keyboard shortcut.
Fcodes	UInts	<p>The list of f-codes that the FrameMaker product executes when the user chooses the menu item or presses the keyboard shortcut.</p> <p>Normally, the first f-code in the list is the same as the f-code specified by the Fcode property.</p>

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

Property name	Data Type	Description
MenuItemType	number	The type of command or menu item. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_MENUITEM_FRAME (1) - the command is a menu item defined by FrameMaker.</li> <li>• Constants.FV_MENUITEM_API (2) - the command is a menu item defined by a client.</li> <li>• Constants.FV_MENUITEM_MACRO (3) - the menu item is not a command; it calls a macro.</li> <li>• Constants.FV_MENUITEM_EXPANDOMATIC (4) - the menu item is an expandomatic menu item (such as !ShowParagraphTags) defined by FrameMaker.</li> </ul>
Mode	number	The mode in which keyboard shortcuts are recorded. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_MODE_MATH (1)</li> <li>• Constants.FV_MODE_NONMATH (2)</li> <li>• Constants.FV_MODE_ALL (3)</li> </ul>
NextCommandInSession	Command	Represents the next command in the list of commands in the session.
ShiftOrUnshiftCommand	Command	If the HasShiftOrUnshiftCommand property is set to Constants.FV_ITEM_HAS_SHIFT_COMMAND (1), the ShiftOrUnshiftCommand property, represents the command to use when the user holds down the Shift key.  If the HasShiftOrUnshiftCommand property is set to Constants.FV_ITEM_HAS_UNSHIFT_COMMAND (2), the ShiftOrUnshiftCommand denotes the command to use when the user is not holding down the Shift key.
Label	string	Indicates the dialog box title.
MenuItemsEnabled	number	Returns 1 if the menu or menu item is enabled. Returns 0 if the menu or menu item is disabled (dimmed).
Name	string	Represents the name of the command, menu, or menu item separator.
NextMenuItemInMenu	FObject	Indicates the next menu item, menu, or separator in the menu.
NextMenuItemInSession	FObject	Specifies the next menu item, menu, or separator in the list of menu items, menus, and separators in the session.
PrevMenuItemInMenu	FObject	Specifies the previous menu item, menu, or separator in the list of menu items, menus, and separators in the session.

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

## Command methods

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

## CompareRet

Property name	Data Type	Description
sum	<a href="#">Doc</a>	Specifies the summary document.
comp	<a href="#">Doc</a>	Specifies the composite document.

## CompareRet methods

[CompareRet](#).



## CondFmt

Property name	Data Type	Description
CondFmtsShown	number	Returns True if the condition is shown. To hide text with a specified condition, set this property and the ShowAll property of the Doc object to False.
Name	string	Denotes the name of the condition format.
NextCondFmtInDoc	CondFmt	Denotes the next condition format in the document (FO_CondFmt ID).
SepOverride	Color	Indicates the color separation format override (FO_Color ID).
StyleOverride	number	The style condition indicators for conditional text. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CN_NO_OVERRIDE (0)</li> <li>• Constants.FV_CN_OVERLINE (1)</li> <li>• Constants.FV_CN_STRIKETHROUGH (2)</li> <li>• Constants.FV_CN_SINGLE_UNDERLINE (3)</li> <li>• Constants.FV_CN_DOUBLE_UNDERLINE (4)</li> <li>• Constants.FV_CN_CHANGEBAR (5)</li> <li>• Constants.FV_CN_NUMERIC_UNDERLINE (6)</li> <li>• Constants.FV_CN_NMRIC_AND_CHNGBAR (7)</li> </ul>
UseSepOverride	number	Returns 1 if color specified by the SepOverride property is used instead of the default color.
BkColor	Color	BkColor is the text background color property whose value is the color to be used.  If this property is applied on a paragraph format, the color is applied as the background color of the whole paragraph.  If this property is applied on a character format, the color is applied as the background color of the text over which the character format is applied.
UseBkColor	bool	The UseBkColor is a Boolean flag which indicates whether the BkColor is set or not. UseBkColor=1 indicates the text has some background color. UseBkColor=0 indicates transparent background.  When UseBkColor=0, the BkColor value is not used.

### CondFmt methods

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

## Constants

Property name	Data Type
Constants.FDK_LIBRARY_VERSION	int <b>Value:</b> 0x11000000
Constants.FE_Success	int <b>Value:</b> 0
Constants.FE_Transport	int <b>Value:</b> -1
Constants.FE_BadDocId	int <b>Value:</b> -2
Constants.FE_BadObjId	int <b>Value:</b> -3
Constants.FE_BadPropNum	int <b>Value:</b> -4
Constants.FE_BadPropType	int <b>Value:</b> -5
Constants.FE_ReadOnly	int <b>Value:</b> -6
Constants.FE_OutOfRange	int <b>Value:</b> -7
Constants.FE_DocModified	int <b>Value:</b> -8
Constants.FE_GroupSelect	int <b>Value:</b> -9
Constants.FE_WithinFrame	int <b>Value:</b> -10
Constants.FE_NotGraphic	int <b>Value:</b> -11
Constants.FE_NotFrame	int <b>Value:</b> -12
Constants.FE_NotGroup	int <b>Value:</b> -13
Constants.FE_BadNewFrame	int <b>Value:</b> -14
Constants.FE_BadNewGroup	int <b>Value:</b> -15

<b>Property name</b>	<b>Data Type</b>
Constants.FE_BadNewSibling	int <b>Value:</b> -16
Constants.FE_BadDelete	int <b>Value:</b> -17
Constants.FE_BadPageDelete	int <b>Value:</b> -18
Constants.FE_TypeUnNamed	int <b>Value:</b> -19
Constants.FE_NameNotFound	int <b>Value:</b> -20
Constants.FE_OffsetNotFound	int <b>Value:</b> -21
Constants.FE_SomeUnresolved	int <b>Value:</b> -22
Constants.FE_BadNew	int <b>Value:</b> -23
Constants.FE_NotBodyPage	int <b>Value:</b> -24
Constants.FE_NotPgf	int <b>Value:</b> -25
Constants.FE_NotBookComponent	int <b>Value:</b> -26
Constants.FE_BadOperation	int <b>Value:</b> -27
Constants.FE_BadElementDefId	int <b>Value:</b> -28
Constants.FE_BadElementId	int <b>Value:</b> -29
Constants.FE_BadNotificationNum	int <b>Value:</b> -30
Constants.FE_BadContainer	int <b>Value:</b> -104
Constants.FE_BadTemplatePath	int <b>Value:</b> -105
Constants.FE_BadXmlApplication	int <b>Value:</b> -106

<b>Property name</b>	<b>Data Type</b>
Constants.FE_DupName	int <b>Value:</b> -32
Constants.FE_BadName	int <b>Value:</b> -33
Constants.FE_CompareTypes	int <b>Value:</b> -34
Constants.FE_BadCompare	int <b>Value:</b> -35
Constants.FE_BadRange	int <b>Value:</b> -36
Constants.FE_PageFrame	int <b>Value:</b> -37
Constants.FE_CantSmooth	int <b>Value:</b> -38
Constants.FE_NotTextFrame	int <b>Value:</b> -39
Constants.FE_HiddenPage	int <b>Value:</b> -40
Constants.FE_NotTextObject	int <b>Value:</b> -41
Constants.FE_SystemError	int <b>Value:</b> -42
Constants.FE_BadParameter	int <b>Value:</b> -43
Constants.FE_Canceled	int <b>Value:</b> -44
Constants.FE_FailedState	int <b>Value:</b> -45
Constants.FE_WantsCustom	int <b>Value:</b> -46
Constants.FE_WantsLandscape	int <b>Value:</b> -47
Constants.FE_WantsPortrait	int <b>Value:</b> -48
Constants.FE_ViewOnly	int <b>Value:</b> -49

<b>Property name</b>	<b>Data Type</b>
Constants.FE_BadSaveFileName	int <b>Value:</b> -50
Constants.FE_GenRuleItemExpected	int <b>Value:</b> -51
Constants.FE_GenRuleMixedConnectors	int <b>Value:</b> -52
Constants.FE_GenRuleLeftBracketExpected	int <b>Value:</b> -53
Constants.FE_GenRuleRightBracketExpected	int <b>Value:</b> -54
Constants.FE_GenRuleAmbiguous	int <b>Value:</b> -55
Constants.FE_GenRuleSyntaxError	int <b>Value:</b> -56
Constants.FE_GenRuleConnectorExpected	int <b>Value:</b> -57
Constants.FE_InvalidString	int <b>Value:</b> -58
Constants.FE_BadSelectionForOperation	int <b>Value:</b> -59
Constants.FE_WrongProduct	int <b>Value:</b> -60
Constants.FE_BookStructured	int <b>Value:</b> -61
Constants.FE_BadRefFlowId	int <b>Value:</b> -62
Constants.FE_FlowStructured	int <b>Value:</b> -63
Constants.FE_BadRefElementId	int <b>Value:</b> -64
Constants.FE_BadInsertPos	int <b>Value:</b> -65
Constants.FE_BadBookId	int <b>Value:</b> -66
Constants.FE_BookUnStructured	int <b>Value:</b> -67

<b>Property name</b>	<b>Data Type</b>
Constants.FE_BadCompPath	int <b>Value:</b> -68
Constants.FE_BadElementSelection	int <b>Value:</b> -69
Constants.FE_FileClosedByClient	int <b>Value:</b> -70
Constants.FE_NotPgOrFlow	int <b>Value:</b> -71
Constants.FE_NotMenu	int <b>Value:</b> -72
Constants.FE_NotCommand	int <b>Value:</b> -73
Constants.FE_NotApiCommand	int <b>Value:</b> -74
Constants.FE_NotInMenu	int <b>Value:</b> -75
Constants.FE_BadShortcut	int <b>Value:</b> -76
Constants.FE_BadMenuBar	int <b>Value:</b> -77
Constants.FE_PropNotSet	int <b>Value:</b> -78
Constants.FE_InvAttributeDef	int <b>Value:</b> -79
Constants.FE_InvAttribute	int <b>Value:</b> -80
Constants.FE_CircularReference	int <b>Value:</b> -81
Constants.FE_NoSuchFlow	int <b>Value:</b> -82
Constants.FE_BadFileType	int <b>Value:</b> -83
Constants.FE_MissingFile	int <b>Value:</b> -84
Constants.FE_CantUpdateMacEdition	int <b>Value:</b> -85

<b>Property name</b>	<b>Data Type</b>
Constants.FE_CanceledByClient	int <b>Value:</b> -86
Constants.FE_EmptyTextObject	int <b>Value:</b> -87
Constants.FE_Busy	int <b>Value:</b> -88
Constants.FE_FilterFailed	int <b>Value:</b> -89
Constants.FE_AsianSystemRequired	int <b>Value:</b> -90
Constants.FE_TintedColor	int <b>Value:</b> -91
Constants.FE_NoColorFamily	int <b>Value:</b> -92
Constants.FE_StringTooLong	int <b>Value:</b> -93
Constants.FE_InternalErrorFailedToWriteInsets	int <b>Value:</b> -94
Constants.FE_NotFound	int <b>Value:</b> -95
Constants.FE_LanguageNotAvailable	int <b>Value:</b> -96
Constants.FE_BadBaseColor	int <b>Value:</b> -97
Constants.FE_BadFamilyName	int <b>Value:</b> -98
Constants.FE_BadInkName	int <b>Value:</b> -99
Constants.FE_ReservedColor	int <b>Value:</b> -100
Constants.FE_TableInLockedTI	int <b>Value:</b> -101
Constants.FE_XRefUnresolved	int <b>Value:</b> -102
Constants.FE_BadXRefSrcDocId	int <b>Value:</b> -103

<b>Property name</b>	<b>Data Type</b>
Constants.FE_InvalidAttrExpr	int <b>Value:</b> -107
Constants.FE_DocAlreadyHasTrackedEdits	int <b>Value:</b> -108
Constants.FE_BookComponentAlreadyHasTrackedEdits	int <b>Value:</b> -109
Constants.FE_AcceptRejectCalledOnWrongPreviewState	int <b>Value:</b> -110
Constants.FE_FailedExportedFileInvalid	int <b>Value:</b> -111
Constants.FE_BadKey	int <b>Value:</b> -177
Constants.FE_BadKeyField	int <b>Value:</b> -178
Constants.FE_BadValue	int <b>Value:</b> -179
Constants.FE_KeyCatalogNotLoaded	int <b>Value:</b> -180
Constants.FE_KeyCatalogIsStale	int <b>Value:</b> -181
Constants.FE_KeyDefinitionDoesNotExist	int <b>Value:</b> -182
Constants.FE_KeyDefinitionAlreadyExists	int <b>Value:</b> -183
Constants.FE_BadFilterType	int <b>Value:</b> -184
Constants.FE_KeyCatalogNotSpecified	int <b>Value:</b> -185
Constants.FE_NonDITADocument	int <b>Value:</b> -186
Constants.FE_UpdateDITAResourceFailedInvalidElement Type	int <b>Value:</b> -187
Constants.FE_UpdateDITAResourceFailedCannotResolve Reference	int <b>Value:</b> -188
Constants.FE_UpdateDITAResourceFailedCannotFindReferencedFile	int <b>Value:</b> -189



**Object Reference**

<b>Property name</b>	<b>Data Type</b>
Constants.FE_UpdateDITAResourceFailedCannotOpenReferencedFile	int <b>Value:</b> -190
Constants.FE_UpdateDITAResourceFailedCannotConvertToFMObject	int <b>Value:</b> -191
Constants.FE_UpdateDITAResourceFailed	int <b>Value:</b> -192
Constants.FE_FDKUndoNotAllowed	int <b>Value:</b> -193
Constants.FR_DialogStayUp	int <b>Value:</b> -10000
Constants.FR_ModalStayUp	int <b>Value:</b> -10000
Constants.FR_CancelOperation	int <b>Value:</b> -10001
Constants.FR_DisplayedTiDialog	int <b>Value:</b> -10002
Constants.FR_ECMImportSuccess	int <b>Value:</b> -10003
Constants.FR_SkipStep	int <b>Value:</b> -10004
Constants.FR_YesOperation	int <b>Value:</b> -10005
Constants.FR_NoOperation	int <b>Value:</b> -10006
Constants.FR_DisplayedXRefDialog	int <b>Value:</b> -10007
Constants.FR_ClosedXRefDialog	int <b>Value:</b> -10008
Constants.FR_DisplayedModelessDialogForNonContainerElem	int <b>Value:</b> -10009
Constants.FR_CancelInsertElementOperation	int <b>Value:</b> -10010
Constants.FR_HideDialogOnClose	int <b>Value:</b> -10011
Constants.FR_DisplayedModalDialogForNonContainerElem	int <b>Value:</b> -10012

<b>Property name</b>	<b>Data Type</b>
Constants.FR_UpdateDITAResponseSuccess	int <b>Value:</b> -10013
Constants.FR_UpdateDITAResponseFailed	int <b>Value:</b> -10014
Constants.FR_UpdateDITAResponseFailedInvalidElement Type	int <b>Value:</b> -10015
Constants.FR_UpdateDITAResponseFailedCannotResolve Reference	int <b>Value:</b> -10016
Constants.FR_UpdateDITAResponseFailedCannotFindRef erencedFile	int <b>Value:</b> -10017
Constants.FR_UpdateDITAResponseFailedCannotOpenRe ferencedFile	int <b>Value:</b> -10018
Constants.FR_UpdateDITAResponseFailedCannotConvert ToFMOobject	int <b>Value:</b> -10019
Constants.FR_KeyCatalogLoaded	int <b>Value:</b> -10020
Constants.FR_CommandEnabled	int <b>Value:</b> -10021
Constants.FR_CommandDisabled	int <b>Value:</b> -10022
Constants.FV_ACCESSBAR_OPEN	int <b>Value:</b> 1
Constants.FV_ACCESSBAR_CLOSE	int <b>Value:</b> 2
Constants.FV_MATH	int <b>Value:</b> 3
Constants.FV_FORMATBAR_OPEN	int <b>Value:</b> 4
Constants.FV_FORMATBAR_CLOSE	int <b>Value:</b> 5
Constants.FV_CHAR_CATALOG	int <b>Value:</b> 6
Constants.FV_PGF_CATALOG	int <b>Value:</b> 7
Constants.FV_ELEMENT_CATALOG	int <b>Value:</b> 8

<b>Property name</b>	<b>Data Type</b>
Constants.FV_GRAPHIC	int <b>Value: 9</b>
Constants.FV_THESAURUS	int <b>Value: 10</b>
Constants.FV_TEMPLATE_BROWSER	int <b>Value: 11</b>
Constants.FV_HIST	int <b>Value: 12</b>
Constants.FV_TBL_CATALOG	int <b>Value: 13</b>
Constants.FV_GRAPHICS_CATALOG	int <b>Value: 14</b>
Constants.FV_HELP_INDEX	int <b>Value: 0</b>
Constants.FV_HELP_KEYS	int <b>Value: 1</b>
Constants.FV_HELP_SAMPLES	int <b>Value: 2</b>
Constants.FV_HELP_OVERVIEW	int <b>Value: 3</b>
Constants.FV_HELP_ONLINE_MANUALS	int <b>Value: 4</b>
Constants.FV_HELP_CONTEXT	int <b>Value: 5</b>
Constants.FV_HELP_INIT_CONTEXT	int <b>Value: 6</b>
Constants.FV_HELP_HYPertext	int <b>Value: 7</b>
Constants.FV_HELP_WEBWORKS	int <b>Value: 8</b>
Constants.FP_DockDialog	int <b>Value: 2344</b>
Constants.FV_DIALOG_DOCK_NONE	int <b>Value: 0x00</b>
Constants.FV_DIALOG_DOCK_LEFT	int <b>Value: 0x1</b>

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

<b>Property name</b>	<b>Data Type</b>
Constants.FP_ResizeGripperPos	int <b>Value:</b> 2350
Constants.FV_DIALOG_RESIZEGRIPPERPOS_NONE	int <b>Value:</b> 0
Constants.FV_DIALOG_RESIZEGRIPPERPOS_FRONT	int <b>Value:</b> 1
Constants.FV_DIALOG_RESIZEGRIPPERPOS_BELOW	int <b>Value:</b> 2
Constants.FP_MinSize	int <b>Value:</b> 2351
Constants.FP_MaxSize	int <b>Value:</b> 2352
Constants.FP_Focus	int <b>Value:</b> 2358
Constants.FP_AlertClientUnique	int <b>Value:</b> 2290
Constants.FP_AlertClientName	int <b>Value:</b> 2291
Constants.FP_AlertString	int <b>Value:</b> 2292
Constants.FP_AlertType	int <b>Value:</b> 2293
Constants.FV_AlertOneButton	int <b>Value:</b> 1
Constants.FV_AlertTwoButton	int <b>Value:</b> 2
Constants.FV_AlertThreeButton	int <b>Value:</b> 3
Constants.FF_ALERT_OK_DEFAULT	int <b>Value:</b> 0
Constants.FF_ALERT_CANCEL_DEFAULT	int <b>Value:</b> 1
Constants.FF_ALERT_CONTINUE_NOTE	int <b>Value:</b> 2
Constants.FF_ALERT_CONTINUE_WARN	int <b>Value:</b> 3

<b>Property name</b>	<b>Data Type</b>
Constants.FF_ALERT_YES_DEFAULT	int <b>Value:</b> 4
Constants.FF_ALERT_NO_DEFAULT	int <b>Value:</b> 5
Constants.FF_ALERT_YES_NO_CANCEL	int <b>Value:</b> 6
Constants.FF_XRUI_INTERNAL	int <b>Value:</b> 0x01
Constants.FF_XRUI_OPEN_DOCS	int <b>Value:</b> 0x02
Constants.FF_XRUI_CLOSED_DOCS	int <b>Value:</b> 0x04
Constants.FF_XRUI_FORCE_UPDATE	int <b>Value:</b> 0x08
Constants.FF_XRUI_EVERYTHING	int <b>Value:</b> (0x01 0x02 0x04)
Constants.FF_Custom_SingleSided	int <b>Value:</b> 0
Constants.FF_Custom_FirstPageRight	int <b>Value:</b> 1
Constants.FF_Custom_FirstPageLeft	int <b>Value:</b> 2
Constants.FF_CLOSE_MODIFIED	int <b>Value:</b> 1
Constants.FF_CMP_SUMMARY_ONLY	int <b>Value:</b> 0x01
Constants.FF_CMP_CHANGE_BARS	int <b>Value:</b> 0x02
Constants.FF_CMP_HYPERLINKS	int <b>Value:</b> 0x04
Constants.FF_CMP_SUMKIT	int <b>Value:</b> 0x08
Constants.FF_CMP_COMPKIT	int <b>Value:</b> 0x10
Constants.FF_CMP_ATTRIBUTES	int <b>Value:</b> 0x20

<b>Property name</b>	<b>Data Type</b>
Constants.FF_IMAGE_BACKGROUND	int <b>Value:</b> 0x0001
Constants.FF_IMAGE_MASTER	int <b>Value:</b> 0x0002
Constants.FF_IMAGE_CLIP	int <b>Value:</b> 0x0004
Constants.FF_IMAGE_NOTRIM	int <b>Value:</b> 0x0008
Constants.FF_IMAGE_PAGEFRAME	int <b>Value:</b> 0x000F
Constants.FV_INTERACT_ROTATE	int <b>Value:</b> 1
Constants.FV_INTERACT_ROTATE_CONSTRAIN	int <b>Value:</b> 2
Constants.FF_ATTACH_BORDER_STATUS	int <b>Value:</b> 0x0001
Constants.FF_ATTACH_BORDER_PAGENUM	int <b>Value:</b> 0x0002
Constants.FF_ATTACH_BORDER_ZOOM	int <b>Value:</b> 0x0004
Constants.FF_ATTACH_BORDER_PAGEUPDOWN	int <b>Value:</b> 0x0008
Constants.FF_ATTACH_BORDER_RIGHTICONS	int <b>Value:</b> 0x0010
Constants.FF_ATTACH_ALL	int <b>Value:</b> (0x0001 0x0002 0x0004 0x0008 0x0010)
Constants.FF_UFF_PGF	int <b>Value:</b> 0x0001
Constants.FF_UFF_FONT	int <b>Value:</b> 0x0002
Constants.FF_UFF_PAGE	int <b>Value:</b> 0x0004
Constants.FF_UFF_TABLE	int <b>Value:</b> 0x0008
Constants.FF_UFF_COND	int <b>Value:</b> 0x0010

<b>Property name</b>	<b>Data Type</b>
Constants.FF_UFF_REFPAGE	int <b>Value:</b> 0x0020
Constants.FF_UFF_VAR	int <b>Value:</b> 0x0040
Constants.FF_UFF_XREF	int <b>Value:</b> 0x0080
Constants.FF_UFF_COLOR	int <b>Value:</b> 0x0100
Constants.FF_UFF_MATH	int <b>Value:</b> 0x0200
Constants.FF_UFF_DOCUMENT_PROPS	int <b>Value:</b> 0x0400
Constants.FF_UFF_COMBINED_FONTS	int <b>Value:</b> 0x0800
Constants.FF_UFF_FBA	int <b>Value:</b> 0x1000
Constants.FF_UFF_REMOVE_PAGE_BREAKS	int <b>Value:</b> 0x4000
Constants.FF_UFF_REMOVE_EXCEPTIONS	int <b>Value:</b> 0x8000
Constants.FF_UFF_ALL	int <b>Value:</b> 0xFFFF
Constants.FF_INTERACTIVE	int <b>Value:</b> 0x0001
Constants.FF_CUT_TBL_CELLS	int <b>Value:</b> 0x0002
Constants.FF_DONT_DELETE_HIDDEN_TEXT	int <b>Value:</b> 0x0004
Constants.FF_INSERT_BELOW_RIGHT	int <b>Value:</b> 0x0008
Constants.FF_VISIBLE_ONLY	int <b>Value:</b> 0x0010
Constants.FF_REPLACE_CELLS	int <b>Value:</b> 0x0020
Constants.FF_DONT_APPLY_ALL_ROWS	int <b>Value:</b> 0x0040



<b>Property name</b>	<b>Data Type</b>
Constants.FF_STRIP_HYPERTEXT	int <b>Value:</b> 0x0080
Constants.FF_SELECT_WHOLE_TABLE	int <b>Value:</b> 0x80000000
Constants.FF_CLEAR_SEL	int <b>Value:</b> 0
Constants.FF_TEXT_SEL	int <b>Value:</b> 1
Constants.FF_TEXT_EXT	int <b>Value:</b> 2
Constants.FF_TEXT_Q_COPY	int <b>Value:</b> 3
Constants.FF_CONTEXT_DRAGTXT	int <b>Value:</b> 4
Constants.FF_PASTE_SEL_TEXT	int <b>Value:</b> 11
Constants.FF_PASTE_CLIP_TEXT	int <b>Value:</b> 12
Constants.FF_COPY_TO_CLIP	int <b>Value:</b> 13
Constants.FF_CELL_SEL	int <b>Value:</b> 21
Constants.FF_CELL_EXT	int <b>Value:</b> 22
Constants.FF_TABLE_SELALL	int <b>Value:</b> 23
Constants.FF_CELL_RES	int <b>Value:</b> 24
Constants.FF_CELLS_RES	int <b>Value:</b> 25
Constants.FF_VIEWER_MENU	int <b>Value:</b> 32
Constants.FF_CONTEXT_MENU	int <b>Value:</b> 33
Constants.FF_STRUCTURE_MENU	int <b>Value:</b> 34

<b>Property name</b>	<b>Data Type</b>
Constants.FF_HYPertext	int <b>Value:</b> 41
Constants.FF_ELEMENT_CONTEXT	int <b>Value:</b> 47
Constants.FF_ELEMENT_EXT_CONTEXT	int <b>Value:</b> 48
Constants.FF_BUBBLE_CONTEXT	int <b>Value:</b> 49
Constants.FF_BUBBLE_EXT_CONTEXT	int <b>Value:</b> 50
Constants.FF_ELEMENT_SEL	int <b>Value:</b> 51
Constants.FF_ELEMENT_EXT	int <b>Value:</b> 52
Constants.FF_BUBBLE_SEL	int <b>Value:</b> 53
Constants.FF_BUBBLE_EXT	int <b>Value:</b> 54
Constants.FF_BUBBLE_COPY	int <b>Value:</b> 55
Constants.FF_COLLAPSE_ELEM	int <b>Value:</b> 56
Constants.FF_COLLAPSE_ALL	int <b>Value:</b> 57
Constants.FF_OPEN_BOOK_COMP	int <b>Value:</b> 58
Constants.FF_PASTE_CLIP_OBJ	int <b>Value:</b> 61
Constants.FF_OBJ_SEL	int <b>Value:</b> 62
Constants.FF_OBJ_EXT	int <b>Value:</b> 63
Constants.FF_OBJ_EXT_BOR	int <b>Value:</b> 64
Constants.FF_OBJ_Q_COPY	int <b>Value:</b> 65

<b>Property name</b>	<b>Data Type</b>
Constants.FF_OBJ_ROTATE	int <b>Value:</b> 66
Constants.FF_OBJ_CROTATE	int <b>Value:</b> 67
Constants.FF_BOR_SEL	int <b>Value:</b> 71
Constants.FF_BOR_EXT	int <b>Value:</b> 72
Constants.FF_CONTEXT_DRAG	int <b>Value:</b> 73
Constants.FF_CONTEXT_DRAG_EXT	int <b>Value:</b> 74
Constants.FF_RES_MOVE	int <b>Value:</b> 81
Constants.FF_RES_MOVE1	int <b>Value:</b> 82
Constants.FF_RES_VERTEX	int <b>Value:</b> 83
Constants.FF_RES_ADD	int <b>Value:</b> 84
Constants.FF_RES_DEL	int <b>Value:</b> 85
Constants.FF_SHIFT_KEY	int <b>Value:</b> 0x0001
Constants.FF_CONTROL_KEY	int <b>Value:</b> 0x0002
Constants.FF_ALT_KEY	int <b>Value:</b> 0x0004
Constants.FF_CMD_KEY	int <b>Value:</b> 0x0010
Constants.FF_ATTRDISP	int <b>Value:</b> 86
Constants.FF_ATTRDISP_ALL	int <b>Value:</b> 87
Constants.FF_EDIT_ATTRIBUTE	int <b>Value:</b> 88

<b>Property name</b>	<b>Data Type</b>
Constants.FF_ATTR_SEL	int <b>Value: 89</b>
Constants.FS_ShowBrowser	int <b>Value: 1</b>
Constants.FS_AlertUserAboutFailure	int <b>Value: 2</b>
Constants.FS_DontNotifyAPIClients	int <b>Value: 3</b>
Constants.FS_UpdateBrowserDirectory	int <b>Value: 4</b>
Constants.FS_MakeVisible	int <b>Value: 5</b>
Constants.FS_Makelconic	int <b>Value: 6</b>
Constants.FS_DisallowMIF	int <b>Value: 7</b>
Constants.FS_DisallowDoc	int <b>Value: 8</b>
Constants.FS_DisallowBookMIF	int <b>Value: 9</b>
Constants.FS_DisallowBookDoc	int <b>Value: 10</b>
Constants.FS_DisallowFilterTypes	int <b>Value: 11</b>
Constants.FS_DisallowPlainText	int <b>Value: 12</b>
Constants.FS_ForceOpenAsText	int <b>Value: 13</b>
Constants.FS_UseRecoverFile	int <b>Value: 14</b>
Constants.FS_UseAutoSaveFile	int <b>Value: 15</b>
Constants.FV_DoCancel	int <b>Value: 0</b>
Constants.FV_DoOK	int <b>Value: 1</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_DoYes	int <b>Value: 2</b>
Constants.FV_DoNo	int <b>Value: 3</b>
Constants.FV_DoShowDialog	int <b>Value: 4</b>
Constants.FV_OpenViewOnly	int <b>Value: 5</b>
Constants.FS_FilesText	int <b>Value: 16</b>
Constants.FV_TextFile_EOLisEOP	int <b>Value: 12</b>
Constants.FV_TextFile_EOLisNotEOP	int <b>Value: 13</b>
Constants.FS_FilesInUse	int <b>Value: 17</b>
Constants.FV_ResetLockAndContinue	int <b>Value: 7</b>
Constants.FV_OpenEditableCopy	int <b>Value: 8</b>
Constants.FS_BooksInUse	int <b>Value: 18</b>
Constants.FS_LockCantBeReset	int <b>Value: 19</b>
Constants.FS_FilesOldVersion	int <b>Value: 20</b>
Constants.FS_FilesStructured	int <b>Value: 21</b>
Constants.FV_StripStructureAndOpen	int <b>Value: 10</b>
Constants.FS_FontNotFoundInDoc	int <b>Value: 22</b>
Constants.FS_FontChangedMetric	int <b>Value: 23</b>
Constants.FS_RefFileNotFound	int <b>Value: 24</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_AllowAllRefFilesUnFindable	int <b>Value: 11</b>
Constants.FS_LanguageNotAvailable	int <b>Value: 25</b>
Constants.FS_OpenInNewWindow	int <b>Value: 26</b>
Constants.FS_OpenId	int <b>Value: 27</b>
Constants.FS_FontNotFoundInCatalog	int <b>Value: 28</b>
Constants.FS_NewDoc	int <b>Value: 29</b>
Constants.FS_OpenDocViewOnly	int <b>Value: 30</b>
Constants.FS_NameStripe	int <b>Value: 31</b>
Constants.FS_BeefyDoc	int <b>Value: 32</b>
Constants.FS_DisallowSgml	int <b>Value: 33</b>
Constants.FS_FileTypeHint	int <b>Value: 34</b>
Constants.FS_UpdateTextReferences	int <b>Value: 35</b>
Constants.FS_UpdateXRefs	int <b>Value: 36</b>
Constants.FS_OpenDocFluid	int <b>Value: 37</b>
Constants.FV_DoUserPreference	int <b>Value: 12</b>
Constants.FS_StructuredOpenApplication	int <b>Value: 38</b>
Constants.FS_SgmlBookFileName	int <b>Value: 39</b>
Constants.FS_OpenFileNotWritable	int <b>Value: 40</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FS_OpenAsType	int <b>Value:</b> 41
Constants.FV_AUTORECOGNIZE	int <b>Value:</b> 0x00
Constants.FV_TYPE_BINARY	int <b>Value:</b> 0x01
Constants.FV_TYPE_MIF	int <b>Value:</b> 0x02
Constants.FV_TYPE_TEXT	int <b>Value:</b> 0x03
Constants.FV_TYPE_SGML	int <b>Value:</b> 0x04
Constants.FV_TYPE_XML	int <b>Value:</b> 0x05
Constants.FV_TYPE_FILTER	int <b>Value:</b> 0xFF
Constants.FS_OpenBookViewOnly	int <b>Value:</b> 42
Constants.FS_DisallowXml	int <b>Value:</b> 43
Constants.FS_NoStructuredErrorLog	int <b>Value:</b> 44
Constants.FS_VerifyInsets	int <b>Value:</b> 45
Constants.FS_IsErrorDoc	int <b>Value:</b> 46
Constants.FS_TemplateShouldInsertRoot	int <b>Value:</b> 47
Constants.FS_CMSId	int <b>Value:</b> 48
Constants.FS_CMSSessionId	int <b>Value:</b> 49
Constants.FS_NumOpenParams	int <b>Value:</b> 49
Constants.FS_OpenedFileName	int <b>Value:</b> 1

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



<b>Property name</b>	<b>Data Type</b>
Constants.FV_FontsMappedInCatalog	int <b>Value: 38</b>
Constants.FV_LanguagesWerentFound	int <b>Value: 39</b>
Constants.FV_BeefyDoc	int <b>Value: 40</b>
Constants.FV_FilesOldVersion	int <b>Value: 41</b>
Constants.FV_FileStructureStripped	int <b>Value: 42</b>
Constants.FV_FilesText	int <b>Value: 43</b>
Constants.FV_OpenedViewOnly	int <b>Value: 44</b>
Constants.FV_EditableCopyOpened	int <b>Value: 45</b>
Constants.FV_BadFileRefsWereMapped	int <b>Value: 46</b>
Constants.FV_ReferencedFilesWerentFound	int <b>Value: 47</b>
Constants.FV_FileAlreadyOpen	int <b>Value: 48</b>
Constants.FV_UnresolvedXRefs	int <b>Value: 49</b>
Constants.FV_UnresolvedTextInsets	int <b>Value: 50</b>
Constants.FV_OpenedFluid	int <b>Value: 51</b>
Constants.FV_FontsWithUnavailableEncodingsUsed	int <b>Value: 53</b>
Constants.FV_FileHadStructure	int <b>Value: 64</b>
Constants.FV_FileAlreadyOpenThisSession	int <b>Value: 65</b>
Constants.FV_BadFileType	int <b>Value: 66</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_BadFileName	int <b>Value:</b> 67
Constants.FV_CantNewBooks	int <b>Value:</b> 68
Constants.FV_CantOpenBooksViewOnly	int <b>Value:</b> 69
Constants.FV_BadScriptValue	int <b>Value:</b> 70
Constants.FV_MissingScript	int <b>Value:</b> 71
Constants.FV_CantForceOpenAsText	int <b>Value:</b> 72
Constants.FV_DisallowedType	int <b>Value:</b> 73
Constants.FV_DocDamagedByTextFilter	int <b>Value:</b> 74
Constants.FV_DocHeadersDamaged	int <b>Value:</b> 75
Constants.FV_DocWrongSize	int <b>Value:</b> 76
Constants.FV_ChecksumDamage	int <b>Value:</b> 77
Constants.FV_CantOpenBooksFluid	int <b>Value:</b> 78
Constants.FV_CancelUseRecoverFile	int <b>Value:</b> 96
Constants.FV_CancelUseAutoSaveFile	int <b>Value:</b> 97
Constants.FV_CancelFileIsText	int <b>Value:</b> 98
Constants.FV_CancelFileIsInUse	int <b>Value:</b> 99
Constants.FV_CancelFileHasStructure	int <b>Value:</b> 100
Constants.FV_CancelReferencedFilesNotFound	int <b>Value:</b> 101

<b>Property name</b>	<b>Data Type</b>
Constants.FV_CancelLanguagesNotFound	int <b>Value:</b> 102
Constants.FV_CancelFontsMapped	int <b>Value:</b> 103
Constants.FV_CancelFontMetricsChanged	int <b>Value:</b> 104
Constants.FV_CancelFontsMappedInCatalog	int <b>Value:</b> 105
Constants.FV_CancelFilesDoc	int <b>Value:</b> 106
Constants.FV_CancelFilesMIF	int <b>Value:</b> 107
Constants.FV_CancelBook	int <b>Value:</b> 108
Constants.FV_CancelBookMIF	int <b>Value:</b> 109
Constants.FV_CancelFilesFilterable	int <b>Value:</b> 110
Constants.FV_CancelFilesOldVersion	int <b>Value:</b> 111
Constants.FV_UserCanceled	int <b>Value:</b> 112
Constants.FV_CancelFileBrowser	int <b>Value:</b> 113
Constants.FV_CancelBeefyDoc	int <b>Value:</b> 114
Constants.FV_CancelFilesSgml	int <b>Value:</b> 115
Constants.FV_CancelFontsWithUnavailableEncodings	int <b>Value:</b> 116
Constants.FV_CancelOpenFileNotWritable	int <b>Value:</b> 117
Constants.FV_CancelTempDiskFull	int <b>Value:</b> 118
Constants.FV_CancelFilesXml	int <b>Value:</b> 119

<b>Property name</b>	<b>Data Type</b>
Constants.FV_TooManyWindows	int <b>Value:</b> 128
Constants.FV_BadTemplate	int <b>Value:</b> 129
Constants.FV_FileNotReadable	int <b>Value:</b> 130
Constants.FS_FileType	int <b>Value:</b> 1
Constants.FV_SaveFmtBinary	int <b>Value:</b> 0
Constants.FV_SaveFmtInterchange	int <b>Value:</b> 1
Constants.FV_SaveFmtStationery	int <b>Value:</b> 3
Constants.FV_SaveFmtViewOnly	int <b>Value:</b> 4
Constants.FV_SaveFmtText	int <b>Value:</b> 6
Constants.FV_SaveFmtSgml	int <b>Value:</b> 7
Constants.FV_SaveFmtFilter	int <b>Value:</b> 8
Constants.FV_SaveFmtPdf	int <b>Value:</b> 9
Constants.FV_SaveFmtXml	int <b>Value:</b> 10
Constants.FV_SaveFmtBinary60	int <b>Value:</b> 11
Constants.FV_SaveFmtBinary70	int <b>Value:</b> 12
Constants.FV_SaveFmtBinary80	int <b>Value:</b> 13
Constants.FV_SaveFmtInterchange70	int <b>Value:</b> 14
Constants.FV_SaveFmtInterchange80	int <b>Value:</b> 15

<b>Property name</b>	<b>Data Type</b>
Constants.FV_SaveFmtBinary90	int <b>Value: 16</b>
Constants.FV_SaveFmtInterchange90	int <b>Value: 17</b>
Constants.FV_SaveFmtCompositeDoc	int <b>Value: 18</b>
Constants.FV_SaveFmtBookWithXml	int <b>Value: 19</b>
Constants.FV_SaveFmtBookWithFm	int <b>Value: 20</b>
Constants.FV_SaveFmtBinary100	int <b>Value: 21</b>
Constants.FV_SaveFmtInterchange100	int <b>Value: 22</b>
Constants.FV_SaveFmtBinary110	int <b>Value: 23</b>
Constants.FV_SaveFmtInterchange110	int <b>Value: 24</b>
Constants.FS_SaveMode	int <b>Value: 4</b>
Constants.FV_ModeSave	int <b>Value: 0</b>
Constants.FV_ModeSaveAs	int <b>Value: 1</b>
Constants.FS_SaveAsModeName	int <b>Value: 5</b>
Constants.FV_SaveAsNameAskUser	int <b>Value: 0</b>
Constants.FV_SaveAsUseFileName	int <b>Value: 1</b>
Constants.FV_SaveAsNameProvided	int <b>Value: 2</b>
Constants.FS_AutoBackupOnSave	int <b>Value: 6</b>
Constants.FV_SaveYesAutoBackup	int <b>Value: 0</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_SaveNoAutoBackup	int <b>Value: 1</b>
Constants.FV_SaveUserPrefAutoBackup	int <b>Value: 2</b>
Constants.FS_MakePageCount	int <b>Value: 7</b>
Constants.FV_UseCurrentSetting	int <b>Value: 0</b>
Constants.FV_DontChangePageCount	int <b>Value: 1</b>
Constants.FV_MakePageCountEven	int <b>Value: 2</b>
Constants.FV_MakePageCountOdd	int <b>Value: 3</b>
Constants.FV_DeleteEmptyPages	int <b>Value: 4</b>
Constants.FS_ShowSaveTextDialog	int <b>Value: 8</b>
Constants.FS_SaveTextTblSetting	int <b>Value: 9</b>
Constants.FV_SaveTblUserPref	int <b>Value: 0</b>
Constants.FV_SaveTblRowsAsPgfs	int <b>Value: 1</b>
Constants.FV_SaveTblColsAsPgfs	int <b>Value: 2</b>
Constants.FV_SaveSkipTbIs	int <b>Value: 3</b>
Constants.FS_SaveTextExtraBlankLineAtEOP	int <b>Value: 10</b>
Constants.FS_UseDefaultUNIXpermissions	int <b>Value: 11</b>
Constants.FS_RetainNameStripe	int <b>Value: 12</b>
Constants.FS_UNIXpermissions	int <b>Value: 13</b>

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

<b>Property name</b>	<b>Data Type</b>
Constants.FV_ProductIsViewer	int <b>Value: 32</b>
Constants.FV_FileNotWritable	int <b>Value: 33</b>
Constants.FV_BadSaveFileName	int <b>Value: 34</b>
Constants.FV_BadFileId	int <b>Value: 35</b>
Constants.FV_BadSaveScriptValue	int <b>Value: 36</b>
Constants.FV_NonPortableSaveName	int <b>Value: 37</b>
Constants.FV_NonPortableFileRefs	int <b>Value: 38</b>
Constants.FV_ProductIsMaker	int <b>Value: 39</b>
Constants.FV_BadSaveObjectId	int <b>Value: 40</b>
Constants.FV_Unstructured	int <b>Value: 41</b>
Constants.FV_InvalidSaveFilter	int <b>Value: 42</b>
Constants.FV_UserCanceledSave	int <b>Value: 48</b>
Constants.FV_FileWasExported	int <b>Value: 49</b>
Constants.FV_CancelSaveFilesInUse	int <b>Value: 50</b>
Constants.FV_CancelSaveFileNotWritable	int <b>Value: 51</b>
Constants.FV_CancelSaveModDateChanged	int <b>Value: 52</b>
Constants.FV_CancelSaveExportedFileInvalid	int <b>Value: 53</b>
Constants.FS_HowToImport	int <b>Value: 4</b>



<b>Property name</b>	<b>Data Type</b>
Constants.FV_DoByRef	int <b>Value: 5</b>
Constants.FV_DoByCopy	int <b>Value: 6</b>
Constants.FV_DoUserChoice	int <b>Value: 7</b>
Constants.FS_ManualUpdate	int <b>Value: 5</b>
Constants.FS_TextInsetName	int <b>Value: 6</b>
Constants.FS_DisallowGraphicTypes	int <b>Value: 9</b>
Constants.FS_DisallowMacEditions	int <b>Value: 10</b>
Constants.FS_FileIsGraphic	int <b>Value: 13</b>
Constants.FS_FitGraphicInSelectedRect	int <b>Value: 14</b>
Constants.FS_GraphicDpi	int <b>Value: 15</b>
Constants.FV_DoImportAsTable	int <b>Value: 8</b>
Constants.FS_ForceImportAsText	int <b>Value: 17</b>
Constants.FS_FileIsMakerDoc	int <b>Value: 18</b>
Constants.FS_UseMainFlow	int <b>Value: 19</b>
Constants.FS_ImportFlowTag	int <b>Value: 20</b>
Constants.FS_ImportFlowPageSpace	int <b>Value: 21</b>
Constants.FV_ReferencePage	int <b>Value: 9</b>
Constants.FV_BodyPage	int <b>Value: 10</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FS_FormatImportedText	int <b>Value: 22</b>
Constants.FV_SourceDoc	int <b>Value: 0</b>
Constants.FV_EnclosingDoc	int <b>Value: 1</b>
Constants.FV_PlainText	int <b>Value: 2</b>
Constants.FS_RemoveManualPageBreaks	int <b>Value: 23</b>
Constants.FS_RemoveOverrides	int <b>Value: 24</b>
Constants.FS_ImportTblTag	int <b>Value: 25</b>
Constants.FS_TblNumHeadingRows	int <b>Value: 26</b>
Constants.FS_LeaveHeadingRowsEmpty	int <b>Value: 27</b>
Constants.FS_TreatParaAsRow	int <b>Value: 28</b>
Constants.FS_CellSeparator	int <b>Value: 29</b>
Constants.FS_NumCellSeparators	int <b>Value: 30</b>
Constants.FS_NumColumns	int <b>Value: 31</b>
Constants.FS_FilesSgmlDoc	int <b>Value: 32</b>
Constants.FS_StructuredImportApplication	int <b>Value: 35</b>
Constants.FS_ImportAsType	int <b>Value: 36</b>
Constants.FS_FilesXmlDoc	int <b>Value: 37</b>
Constants.FS_RasterDpi	int <b>Value: 38</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FS_RasterImageWidth	int <b>Value: 39</b>
Constants.FS_RasterImageHeight	int <b>Value: 40</b>
Constants.FS_ShowRasterDpiDialog	int <b>Value: 41</b>
Constants.FS_ClientUpdateSilent	int <b>Value: 47</b>
Constants.FS_NumImportParams	int <b>Value: 47</b>
Constants.FS_InsetData	int <b>Value: 50</b>
Constants.FS_UseHTTP	int <b>Value: 51</b>
Constants.FS_InsetElementDef	int <b>Value: 52</b>
Constants.FS_PDFPageNum	int <b>Value: 45</b>
Constants.FS_ImportedFileName	int <b>Value: 1</b>
Constants.FS_ImportNativeError	int <b>Value: 2</b>
Constants.FS_ImportStatus	int <b>Value: 3</b>
Constants.FS_NumImportReturnParams	int <b>Value: 3</b>
Constants.FV_NumImportStatusFields	int <b>Value: 2</b>
Constants.FV_ImportedByCopy	int <b>Value: 0</b>
Constants.FV_ImportedText	int <b>Value: 1</b>
Constants.FV_ImportedTextTable	int <b>Value: 2</b>
Constants.FV_ImportedMIF	int <b>Value: 3</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_ImportedMakerDoc	int <b>Value: 4</b>
Constants.FV_ImportedFilteredFile	int <b>Value: 5</b>
Constants.FV_ImportedGraphicFile	int <b>Value: 6</b>
Constants.FV_ImportedMacEdition	int <b>Value: 7</b>
Constants.FV_ImportedSgmlDoc	int <b>Value: 8</b>
Constants.FV_ImportedXmlDoc	int <b>Value: 9</b>
Constants.FV_BadImportFileName	int <b>Value: 16</b>
Constants.FV_BadImportFileType	int <b>Value: 17</b>
Constants.FV_BadImportScriptValue	int <b>Value: 18</b>
Constants.FV_MissingImportScript	int <b>Value: 19</b>
Constants.FV_CantForceImportAsText	int <b>Value: 20</b>
Constants.FV_DisallowedImportType	int <b>Value: 21</b>
Constants.FV_NoMainFlow	int <b>Value: 22</b>
Constants.FV_NoFlowWithSpecifiedName	int <b>Value: 23</b>
Constants.FV_InsertionPointNotInText	int <b>Value: 24</b>
Constants.FV_InsertionPointInTableCell	int <b>Value: 25</b>
Constants.FV_InsertionPointInFootnote	int <b>Value: 26</b>
Constants.FV_InsufficientMemory	int <b>Value: 27</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_BadEnclosingDocId	int <b>Value: 28</b>
Constants.FV_BadTextFileTypeHint	int <b>Value: 29</b>
Constants.FV_FlowUnstructured	int <b>Value: 30</b>
Constants.FV_CancelFileText	int <b>Value: 32</b>
Constants.FV_CancelFileDoc	int <b>Value: 33</b>
Constants.FV_CancelFileMIF	int <b>Value: 34</b>
Constants.FV_CancelFileFilterable	int <b>Value: 35</b>
Constants.FV_CancelFileGraphic	int <b>Value: 36</b>
Constants.FV_UserCanceledImport	int <b>Value: 37</b>
Constants.FV_CancelImportBrowser	int <b>Value: 38</b>
Constants.FV_CancelFileIsMacEdition	int <b>Value: 39</b>
Constants.FV_CancelFileSgml	int <b>Value: 40</b>
Constants.FV_CancelFileXml	int <b>Value: 41</b>
Constants.FV_ImportFileNotReadable	int <b>Value: 48</b>
Constants.FS_Export	int <b>Value: 4</b>
Constants.FV_ExportDocument	int <b>Value: 0</b>
Constants.FV_ExportObject	int <b>Value: 1</b>
Constants.FS_ExportType	int <b>Value: 5</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FS_ExportTextRange	int <b>Value: 6</b>
Constants.FS_ExportObjectId	int <b>Value: 7</b>
Constants.FS_ExportObjectNeedsDpi	int <b>Value: 12</b>
Constants.FS_ExportFileTypeHint	int <b>Value: 14</b>
Constants.FS_StructuredExportApplication	int <b>Value: 16</b>
Constants.FS_ImportExportVersion	int <b>Value: 17</b>
Constants.FS_NumExportParams	int <b>Value: 17</b>
Constants.FS_ExportFileName	int <b>Value: 1</b>
Constants.FS_ExportNativeError	int <b>Value: 2</b>
Constants.FS_ExportStatus	int <b>Value: 3</b>
Constants.FV_NumExportStatusFields	int <b>Value: 2</b>
Constants.FS_NumExportReturnParams	int <b>Value: 3</b>
Constants.FV_ExportFileHasNewName	int <b>Value: 0</b>
Constants.FV_ExportProductIsViewer	int <b>Value: 1</b>
Constants.FV_ExportFileNotWritable	int <b>Value: 2</b>
Constants.FV_InsufficientMem	int <b>Value: 3</b>
Constants.FV_BadExportFileName	int <b>Value: 16</b>
Constants.FV_MissingExportScript	int <b>Value: 17</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_BadExportObjectId	int <b>Value: 18</b>
Constants.FV_InvalidTextRange	int <b>Value: 19</b>
Constants.FV_BadExportDocId	int <b>Value: 20</b>
Constants.FV_BadExportScriptValue	int <b>Value: 21</b>
Constants.FV_FilterFailed	int <b>Value: 22</b>
Constants.FV_InvalidExportFilter	int <b>Value: 23</b>
Constants.FV_UserCanceledExport	int <b>Value: 32</b>
Constants.FV_CanceledExportObjectNeedsDpi	int <b>Value: 33</b>
Constants.FV_ApiClientCanceledExport	int <b>Value: 34</b>
Constants.FS_AllowNonFMFiles	int <b>Value: 1</b>
Constants.FS_AllowViewOnlyFiles	int <b>Value: 3</b>
Constants.FS_ShowBookErrorLog	int <b>Value: 4</b>
Constants.FS_AllowInconsistentNumProps	int <b>Value: 6</b>
Constants.FS_UpdateBookGeneratedFiles	int <b>Value: 7</b>
Constants.FS_UpdateBookNumbering	int <b>Value: 8</b>
Constants.FS_UpdateBookOleLinks	int <b>Value: 9</b>
Constants.FS_UpdateBookTextReferences	int <b>Value: 10</b>
Constants.FS_UpdateBookXRefs	int <b>Value: 11</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FS_UpdateBookMasterPages	int <b>Value: 12</b>
Constants.FS_NumUpdateBookParams	int <b>Value: 12</b>
Constants.FS_UpdateBookStatus	int <b>Value: 1</b>
Constants.FS_NumUpdateBookReturnParams	int <b>Value: 1</b>
Constants.FV_NumUpdateBookStatusFields	int <b>Value: 4</b>
Constants.FV_BookNotSelfConsistent	int <b>Value: 0</b>
Constants.FV_DuplicateFileInBook	int <b>Value: 1</b>
Constants.FV_NoNonGeneratedFilesInBook	int <b>Value: 2</b>
Constants.FV_CancelNonFMFileInBook	int <b>Value: 32</b>
Constants.FV_CancelViewOnlyFileInBook	int <b>Value: 33</b>
Constants.FV_CancelInconsistentNumPropsInFileInBook	int <b>Value: 34</b>
Constants.FV_UserCanceledUpdateBook	int <b>Value: 35</b>
Constants.FV_BadUpdateBookFileId	int <b>Value: 64</b>
Constants.FV_BadUpdateBookScriptValue	int <b>Value: 65</b>
Constants.FV_FileInBookNotOpened	int <b>Value: 96</b>
Constants.FV_FileInBookNotSaved	int <b>Value: 97</b>
Constants.FV_TooManyWindowsUpdateBook	int <b>Value: 98</b>
Constants.FV_Body	int <b>Value: 1237</b>



<b>Property name</b>	<b>Data Type</b>
Constants.FV_Heading	int <b>Value:</b> 1238
Constants.FV_Footing	int <b>Value:</b> 1239
Constants.FV_Above	int <b>Value:</b> 1240
Constants.FV_Below	int <b>Value:</b> 1241
Constants.FV_Left	int <b>Value:</b> 1242
Constants.FV_Right	int <b>Value:</b> 1243
Constants.FV_ChooseSelect	int <b>Value:</b> 0x00000000
Constants.FV_ChooseOpen	int <b>Value:</b> 0x00000001
Constants.FV_ChooseSave	int <b>Value:</b> 0x00000002
Constants.FV_ChooseOpenDir	int <b>Value:</b> 0x00000004
Constants.FV_ChooseMultiSelect	int <b>Value:</b> 0x00000008
Constants.FV_ChooseMultiOpen	int <b>Value:</b> 0x00000010
Constants.FV_BrowseLocalOnly	int <b>Value:</b> 0x00010000
Constants.FV_BrowseCmsOnly	int <b>Value:</b> 0x00020000
Constants.FV_BrowseLocalWithWebdavOnly	int <b>Value:</b> 0x00040000
Constants.FA_Init_First	int <b>Value:</b> 1
Constants.FA_Init_Subsequent	int <b>Value:</b> 2
Constants.FA_Init_TakeControl	int <b>Value:</b> 3

<b>Property name</b>	<b>Data Type</b>
Constants.FA_Init_DocReport	int <b>Value: 4</b>
Constants.FA_Note_PreOpenDoc	int <b>Value: 1</b>
Constants.FA_Note_PostOpenDoc	int <b>Value: 2</b>
Constants.FA_Note_PreOpenMIF	int <b>Value: 3</b>
Constants.FA_Note_PostOpenMIF	int <b>Value: 4</b>
Constants.FA_Note_PreSaveDoc	int <b>Value: 5</b>
Constants.FA_Note_PostSaveDoc	int <b>Value: 6</b>
Constants.FA_Note_PreSaveMIF	int <b>Value: 7</b>
Constants.FA_Note_PostSaveMIF	int <b>Value: 8</b>
Constants.FA_Note_PreFileType	int <b>Value: 9</b>
Constants.FA_Note_PostFileType	int <b>Value: 10</b>
Constants.FA_Note_PreQuitDoc	int <b>Value: 11</b>
Constants.FA_Note_DirtyDoc	int <b>Value: 12</b>
Constants.FA_Note_ClientCall	int <b>Value: 13</b>
Constants.FA_Note_FilterIn	int <b>Value: 14</b>
Constants.FA_Note_FilterOut	int <b>Value: 15</b>
Constants.FA_Note_PreOpenBook	int <b>Value: 16</b>
Constants.FA_Note_PostOpenBook	int <b>Value: 17</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FA_Note_PreOpenBookMIF	int <b>Value: 18</b>
Constants.FA_Note_PostOpenBookMIF	int <b>Value: 19</b>
Constants.FA_Note_PreSaveBook	int <b>Value: 20</b>
Constants.FA_Note_PostSaveBook	int <b>Value: 21</b>
Constants.FA_Note_PreSaveBookMIF	int <b>Value: 22</b>
Constants.FA_Note_PostSaveBookMIF	int <b>Value: 23</b>
Constants.FA_Note_PreQuitBook	int <b>Value: 24</b>
Constants.FA_Note_DirtyBook	int <b>Value: 25</b>
Constants.FA_Note_PreQuitSession	int <b>Value: 26</b>
Constants.FA_Note_PostQuitSession	int <b>Value: 27</b>
Constants.FA_Note_PreRevertDoc	int <b>Value: 28</b>
Constants.FA_Note_PostRevertDoc	int <b>Value: 29</b>
Constants.FA_Note_PreRevertBook	int <b>Value: 30</b>
Constants.FA_Note_PostRevertBook	int <b>Value: 31</b>
Constants.FA_Note_PreAutoSaveDoc	int <b>Value: 32</b>
Constants.FA_Note_PostAutoSaveDoc	int <b>Value: 33</b>
Constants.FA_Note_BackToUser	int <b>Value: 34</b>
Constants.FA_Note_DisplayClientTiDialog	int <b>Value: 35</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FA_Note_UpdateAllClientTi	int <b>Value: 36</b>
Constants.FA_Note_UpdateClientTi	int <b>Value: 37</b>
Constants.FA_Note_PreImport	int <b>Value: 38</b>
Constants.FA_Note_PostImport	int <b>Value: 39</b>
Constants.FA_Note_PostQuitDoc	int <b>Value: 40</b>
Constants.FA_Note_PostQuitBook	int <b>Value: 41</b>
Constants.FA_Note_PreFunction	int <b>Value: 42</b>
Constants.FA_Note_PostFunction	int <b>Value: 43</b>
Constants.FA_Note_PreMouseCommand	int <b>Value: 44</b>
Constants.FA_Note_PostMouseCommand	int <b>Value: 45</b>
Constants.FA_Note_PreHypertext	int <b>Value: 46</b>
Constants.FA_Note_PostHypertext	int <b>Value: 47</b>
Constants.FA_Note_PrePrint	int <b>Value: 48</b>
Constants.FA_Note_PostPrint	int <b>Value: 49</b>
Constants.FA_Note_BodyPageAdded	int <b>Value: 50</b>
Constants.FA_Note_BodyPageDeleted	int <b>Value: 51</b>
Constants.FA_Note_PreInsertElement	int <b>Value: 52</b>
Constants.FA_Note_PostInsertElement	int <b>Value: 53</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FA_Note_PreChangeElement	int <b>Value: 54</b>
Constants.FA_Note_PostChangeElement	int <b>Value: 55</b>
Constants.FA_Note_PreWrapElement	int <b>Value: 56</b>
Constants.FA_Note_PostWrapElement	int <b>Value: 57</b>
Constants.FA_Note_PreDragElement	int <b>Value: 58</b>
Constants.FA_Note_PostDragElement	int <b>Value: 59</b>
Constants.FA_Note_PreCopyElement	int <b>Value: 60</b>
Constants.FA_Note_PostCopyElement	int <b>Value: 61</b>
Constants.FA_Note_PreSetAttrValue	int <b>Value: 62</b>
Constants.FA_Note_PostSetAttrValue	int <b>Value: 63</b>
Constants.FA_Note_PreImportElemDefs	int <b>Value: 64</b>
Constants.FA_Note_PostImportElemDefs	int <b>Value: 65</b>
Constants.FA_Note_ECMInternal	int <b>Value: 66</b>
Constants.FA_Note_PreExport	int <b>Value: 67</b>
Constants.FA_Note_PostExport	int <b>Value: 68</b>
Constants.FA_Note_PreInlineTypeIn	int <b>Value: 69</b>
Constants.FA_Note_PostInlineTypeIn	int <b>Value: 70</b>
Constants.FA_Note_PreSaveAsPDFDialog	int <b>Value: 71</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FA_Note_PostSaveAsPDFDialog	int <b>Value: 72</b>
Constants.FA_Note_PreDistill	int <b>Value: 73</b>
Constants.FA_Note_PostDistill	int <b>Value: 74</b>
Constants.FA_Note_FilterFileToFile	int <b>Value: 75</b>
Constants.FA_Note_PreBookComponentOpen	int <b>Value: 76</b>
Constants.FA_Note_PostBookComponentOpen	int <b>Value: 77</b>
Constants.FA_Note_PreGenerate	int <b>Value: 78</b>
Constants.FA_Note_PostGenerate	int <b>Value: 79</b>
Constants.FA_Note_PreGoToXrefSrc	int <b>Value: 80</b>
Constants.FA_Note_PostGoToXrefSrc	int <b>Value: 81</b>
Constants.FA_Note_PreOpenSGML	int <b>Value: 82</b>
Constants.FA_Note_PostOpenSGML	int <b>Value: 83</b>
Constants.FA_Note_Dialog	int <b>Value: 84</b>
Constants.FA_Note_Alert	int <b>Value: 85</b>
Constants.FA_Note_Palette	int <b>Value: 86</b>
Constants.FA_Note_ToolBar	int <b>Value: 87</b>
Constants.FA_Note_ConsoleMessage	int <b>Value: 88</b>
Constants.FA_Note_Help	int <b>Value: 89</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FA_Note_URL	int <b>Value: 90</b>
Constants.FA_Note_CursorChange	int <b>Value: 91</b>
Constants.FA_Note_FontSubstitution	int <b>Value: 92</b>
Constants.FA_Note_UndoCheckpoint	int <b>Value: 93</b>
Constants.FA_Note_FileOpen	int <b>Value: 94</b>
Constants.FA_Note_PreOpenXML	int <b>Value: 95</b>
Constants.FA_Note_PostOpenXML	int <b>Value: 96</b>
Constants.FA_Note_PreSaveXML	int <b>Value: 97</b>
Constants.FA_Note_PostSaveXML	int <b>Value: 98</b>
Constants.FA_Note_PreSaveSGML	int <b>Value: 99</b>
Constants.FA_Note_PostSaveSGML	int <b>Value: 100</b>
Constants.FA_Note_U3DCommand	int <b>Value: 101</b>
Constants.FA_Note_Not_U3DCommand	int <b>Value: 102</b>
Constants.FA_Note_Not_RSC_Supported_File	int <b>Value: 103</b>
Constants.FA_Note_RSC_Supported_File	int <b>Value: 104</b>
Constants.FA_Note_PostActiveDocChange	int <b>Value: 105</b>
Constants.FA_Note_PreUpdateXRefs	int <b>Value: 106</b>
Constants.FA_Note_PostUpdateXRefs	int <b>Value: 107</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FA_Note_DisplayClientXRefDialog	int <b>Value:</b> 108
Constants.FA_Note_QuitModelessDialog	int <b>Value:</b> 109
Constants.FA_Note_InsertTopicRef	int <b>Value:</b> 110
Constants.FA_Note_InsertConRef	int <b>Value:</b> 111
Constants.FA_Note_GenerateFM	int <b>Value:</b> 112
Constants.FA_Note_OpenAllTopicrefs	int <b>Value:</b> 113
Constants.FA_Note_UpdateRefs	int <b>Value:</b> 114
Constants.FA_Note_AssignId	int <b>Value:</b> 115
Constants.FA_Note_DITAOptions	int <b>Value:</b> 116
Constants.FA_Note_NewDitamapFile	int <b>Value:</b> 117
Constants.FA_Note_NewBookmapFile	int <b>Value:</b> 118
Constants.FA_Note_NewTopicFile	int <b>Value:</b> 119
Constants.FA_Note_NewTaskFile	int <b>Value:</b> 120
Constants.FA_Note_NewConceptFile	int <b>Value:</b> 121
Constants.FA_Note_NewReferenceFile	int <b>Value:</b> 122
Constants.FA_Note_NewGlossEntryFile	int <b>Value:</b> 123
Constants.FA_Note_SWF_File	int <b>Value:</b> 124
Constants.FA_Note_Not_SWF_File	int <b>Value:</b> 125



<b>Property name</b>	<b>Data Type</b>
Constants.FA_Note_Enable_Disable_DITA_Menu_Commands	int <b>Value:</b> 126
Constants.FA_Note_Dialog_Create	int <b>Value:</b> 127
Constants.FA_Note_SetAttrValue	int <b>Value:</b> 128
Constants.FA_Note_UpdateRefs_On_Save	int <b>Value:</b> 129
Constants.FA_Note_Read_Struct_App	int <b>Value:</b> 130
Constants.FA_Note_Struct_Wizard	int <b>Value:</b> 131
Constants.FA_Note_Open_Dtd	int <b>Value:</b> 132
Constants.FA_Note_Open_Schema	int <b>Value:</b> 133
Constants.FA_Note_Open_EDD_For_App	int <b>Value:</b> 134
Constants.FA_Note_PostSetPropertyValue	int <b>Value:</b> 135
Constants.FA_Note_Not_AI_Supported_File	int <b>Value:</b> 136
Constants.FA_Note_AI_Supported_File	int <b>Value:</b> 137
Constants.FA_Note_Poster_Applied	int <b>Value:</b> 138
Constants.FA_Note_PrePublishDitamap	int <b>Value:</b> 139
Constants.FA_Note_PostPublishDitamap	int <b>Value:</b> 140
Constants.FA_Note_UpdateDITAReference	int <b>Value:</b> 141
Constants.FA_Note_UpdateDITAReferences	int <b>Value:</b> 142
Constants.FA_Note_LoadKeyCatalog	int <b>Value:</b> 143

<b>Property name</b>	<b>Data Type</b>
Constants.FA_Note_ReLoadKeyCatalog	int <b>Value:</b> 144
Constants.FA_Note_EnumeratedComponent	int <b>Value:</b> 145
Constants.FA_Note_PreSwitchView	int <b>Value:</b> 146
Constants.FA_Note_PostSwitchView	int <b>Value:</b> 147
Constants.FA_Note_IsCommandEnabled	int <b>Value:</b> 148
Constants.FA_Note_Num	int <b>Value:</b> 149
Constants.FTI_String	int <b>Value:</b> 0x00000001
Constants.FTI_LineBegin	int <b>Value:</b> 0x00000002
Constants.FTI_LineEnd	int <b>Value:</b> 0x00000004
Constants.FTI_PgfBegin	int <b>Value:</b> 0x00000008
Constants.FTI_PgfEnd	int <b>Value:</b> 0x00000010
Constants.FTI_FlowBegin	int <b>Value:</b> 0x00000020
Constants.FTI_FlowEnd	int <b>Value:</b> 0x00000040
Constants.FTI_PageBegin	int <b>Value:</b> 0x00000080
Constants.FTI_PageEnd	int <b>Value:</b> 0x00000100
Constants.FTI_SubColBegin	int <b>Value:</b> 0x00000200
Constants.FTI_SubColEnd	int <b>Value:</b> 0x00000400
Constants.FTI_FrameAnchor	int <b>Value:</b> 0x00000800

<b>Property name</b>	<b>Data Type</b>
Constants.FTI_FnAnchor	int <b>Value:</b> 0x00001000
Constants.FTI_TblAnchor	int <b>Value:</b> 0x00002000
Constants.FTI_MarkerAnchor	int <b>Value:</b> 0x00004000
Constants.FTI_XRefBegin	int <b>Value:</b> 0x00008000
Constants.FTI_XRefEnd	int <b>Value:</b> 0x00010000
Constants.FTI_VarBegin	int <b>Value:</b> 0x00020000
Constants.FTI_VarEnd	int <b>Value:</b> 0x00040000
Constants.FTI_ElementBegin	int <b>Value:</b> 0x00080000
Constants.FTI_ElementEnd	int <b>Value:</b> 0x00100000
Constants.FTI_CharPropsChange	int <b>Value:</b> 0x00200000
Constants.FTI_TextFrameBegin	int <b>Value:</b> 0x00400000
Constants.FTI_TextFrameEnd	int <b>Value:</b> 0x00800000
Constants.FTI_TextObjId	int <b>Value:</b> 0x01000000
Constants.FTI_TextInsetBegin	int <b>Value:</b> 0x02000000
Constants.FTI_TextInsetEnd	int <b>Value:</b> 0x04000000
Constants.FTI_ElemPrefixBegin	int <b>Value:</b> 0x08000000
Constants.FTI_ElemPrefixEnd	int <b>Value:</b> 0x10000000
Constants.FTI_ElemSuffixBegin	int <b>Value:</b> 0x20000000

<b>Property name</b>	<b>Data Type</b>
Constants.FTI_ElemSuffixEnd	int <b>Value:</b> 0x40000000
Constants.FTI2_RubiTextBegin	int <b>Value:</b> 0x80000001
Constants.FTI2_RubiTextEnd	int <b>Value:</b> 0x80000002
Constants.FTI2_RubiCompositeBegin	int <b>Value:</b> 0x80000004
Constants.FTI2_RubiCompositeEnd	int <b>Value:</b> 0x80000008
Constants.FTI_HardLineEnd	int <b>Value:</b> 1
Constants.FTI_HyphenLineEnd	int <b>Value:</b> 2
Constants.FTF_FAMILY	int <b>Value:</b> 0x80000000
Constants.FTF_VARIATION	int <b>Value:</b> 0x40000000
Constants.FTF_WEIGHT	int <b>Value:</b> 0x20000000
Constants.FTF_ANGLE	int <b>Value:</b> 0x10000000
Constants.FTF_UNDERLINING	int <b>Value:</b> 0x08000000
Constants.FTF_STRIKETHROUGH	int <b>Value:</b> 0x04000000
Constants.FTF_OVERLINE	int <b>Value:</b> 0x02000000
Constants.FTF_CHANGEBAR	int <b>Value:</b> 0x01000000
Constants.FTF_OUTLINE	int <b>Value:</b> 0x00800000
Constants.FTF_SHADOW	int <b>Value:</b> 0x00400000
Constants.FTF_PAIRKERN	int <b>Value:</b> 0x00200000

<b>Property name</b>	<b>Data Type</b>
Constants.FTF_SIZE	int <b>Value:</b> 0x00100000
Constants.FTF_KERNX	int <b>Value:</b> 0x00080000
Constants.FTF_KERNY	int <b>Value:</b> 0x00040000
Constants.FTF_SPREAD	int <b>Value:</b> 0x00020000
Constants.FTF_COLOR	int <b>Value:</b> 0x00010000
Constants.FTF_CHARTAG	int <b>Value:</b> 0x00008000
Constants.FTF_CAPITALIZATION	int <b>Value:</b> 0x00004000
Constants.FTF_POSITION	int <b>Value:</b> 0x00002000
Constants.FTF_STRETCH	int <b>Value:</b> 0x00001000
Constants.FTF_LANGUAGE	int <b>Value:</b> 0x00000800
Constants.FTF_TSUME	int <b>Value:</b> 0x00000400
Constants.FTF_ENCODING	int <b>Value:</b> 0x00000200
Constants.FTF_IIF	int <b>Value:</b> 0x00000002
Constants.FTF_CONDITIONTAG	int <b>Value:</b> 0x00000001
Constants.FTF_ALL	int <b>Value:</b> 0xFFFFFE03
Constants.FV_OBJ_END_OFFSET	int <b>Value:</b> 0x50000000
Constants.FT_Bad	int <b>Value:</b> 0
Constants.FT_Integer	int <b>Value:</b> 1

<b>Property name</b>	<b>Data Type</b>
Constants.FT_Metric	int <b>Value: 2</b>
Constants.FT_String	int <b>Value: 3</b>
Constants.FT_Id	int <b>Value: 4</b>
Constants.FT_Metrics	int <b>Value: 5</b>
Constants.FT_Strings	int <b>Value: 6</b>
Constants.FT_Points	int <b>Value: 7</b>
Constants.FT_Tabs	int <b>Value: 8</b>
Constants.FT_TextLoc	int <b>Value: 9</b>
Constants.FT_TextRange	int <b>Value: 10</b>
Constants.oldFT_ElementFmts	int <b>Value: 11</b>
Constants.FT_ElementCatalog	int <b>Value: 12</b>
Constants.FT_Ints	int <b>Value: 13</b>
Constants.FT_UBytes	int <b>Value: 14</b>
Constants.FT_UInts	int <b>Value: 15</b>
Constants.FT_FormatBranches	int <b>Value: 16</b>
Constants.FT_AttributeDefs	int <b>Value: 17</b>
Constants.FT_Attributes	int <b>Value: 18</b>
Constants.FT_ElementRange	int <b>Value: 19</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FT_Vals	int <b>Value: 20</b>
Constants.FT_AttributesEx	int <b>Value: 21</b>
Constants.FT_Num	int <b>Value: 22</b>
Constants.FI_PLUGIN_PRODUCTNAME	int <b>Value: 1</b>
Constants.FI_PLUGIN_NAME	int <b>Value: 2</b>
Constants.FI_PLUGIN_TYPE	int <b>Value: 3</b>
Constants.FI_PLUGIN_DESCRIPTION	int <b>Value: 4</b>
Constants.FI_PLUGIN_PRODUCTS	int <b>Value: 5</b>
Constants.FI_PLUGIN_FACET	int <b>Value: 6</b>
Constants.FI_PLUGIN_FORMATID	int <b>Value: 7</b>
Constants.FI_PLUGIN_VENDOR	int <b>Value: 8</b>
Constants.FI_PLUGIN_SUFFIX	int <b>Value: 9</b>
Constants.FI_PLUGIN_INFORMAT	int <b>Value: 10</b>
Constants.FI_PLUGIN_OUTFORMAT	int <b>Value: 11</b>
Constants.FO_Session	int <b>Value: 0</b>
Constants.FO_StringResource	int <b>Value: 1</b>
Constants.FO_Book	int <b>Value: 2</b>
Constants.FO_BookComponent	int <b>Value: 3</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FO_Doc	int <b>Value: 4</b>
Constants.FO_BodyPage	int <b>Value: 5</b>
Constants.FO_MasterPage	int <b>Value: 6</b>
Constants.FO_RefPage	int <b>Value: 7</b>
Constants.FO_HiddenPage	int <b>Value: 8</b>
Constants.FO_Last_Page	int <b>Value: 8</b>
Constants.FO_UnanchoredFrame	int <b>Value: 9</b>
Constants.FO_Group	int <b>Value: 10</b>
Constants.FO_Arc	int <b>Value: 11</b>
Constants.FO_Rectangle	int <b>Value: 12</b>
Constants.FO_Ellipse	int <b>Value: 13</b>
Constants.FO_RoundRect	int <b>Value: 14</b>
Constants.FO_Polyline	int <b>Value: 15</b>
Constants.FO_Polygon	int <b>Value: 16</b>
Constants.FO_Line	int <b>Value: 17</b>
Constants.FO_TextLine	int <b>Value: 18</b>
Constants.FO_TextFrame	int <b>Value: 19</b>
Constants.FO_Inset	int <b>Value: 20</b>



<b>Property name</b>	<b>Data Type</b>
Constants.FO_Math	int <b>Value: 21</b>
Constants.FO_DBGroup	int <b>Value: 22</b>
Constants.FO_Last_Graphic	int <b>Value: 22</b>
Constants.FO_AFrame	int <b>Value: 23</b>
Constants.FO_XLast_Graphic	int <b>Value: 23</b>
Constants.FO_Marker	int <b>Value: 24</b>
Constants.FO_Fn	int <b>Value: 25</b>
Constants.FO_XRef	int <b>Value: 26</b>
Constants.FO_XRefFmt	int <b>Value: 27</b>
Constants.FO_Var	int <b>Value: 28</b>
Constants.FO_VarFmt	int <b>Value: 29</b>
Constants.FO_Flow	int <b>Value: 30</b>
Constants.FO_Pgf	int <b>Value: 31</b>
Constants.FO_PgfFmt	int <b>Value: 32</b>
Constants.FO_CharFmt	int <b>Value: 33</b>
Constants.FO_CondFmt	int <b>Value: 34</b>
Constants.FO_Color	int <b>Value: 35</b>
Constants.FO_Tbl	int <b>Value: 36</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FO_TblFmt	int <b>Value: 37</b>
Constants.FO_Row	int <b>Value: 38</b>
Constants.FO_Cell	int <b>Value: 39</b>
Constants.FO_RulingFmt	int <b>Value: 40</b>
Constants.FO_ElementDef	int <b>Value: 41</b>
Constants.FO_Element	int <b>Value: 42</b>
Constants.FO_DialogResource	int <b>Value: 43</b>
Constants.FO_DlgBox	int <b>Value: 44</b>
Constants.FO_DlgButton	int <b>Value: 45</b>
Constants.FO_DlgTriBox	int <b>Value: 46</b>
Constants.FO_DlgCheckBox	int <b>Value: 47</b>
Constants.FO_DlgPopUp	int <b>Value: 48</b>
Constants.FO_DlgScrollBar	int <b>Value: 49</b>
Constants.FO_DlgEditBox	int <b>Value: 50</b>
Constants.FO_DlgRadioButton	int <b>Value: 51</b>
Constants.FO_DlgLabel	int <b>Value: 52</b>
Constants.FO_DlgImage	int <b>Value: 53</b>
Constants.FO_DlgScrollBar	int <b>Value: 54</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FO_Menu	int <b>Value: 55</b>
Constants.FO_Command	int <b>Value: 56</b>
Constants.FO_MenultemSeparator	int <b>Value: 57</b>
Constants.FO_FmtChangeList	int <b>Value: 58</b>
Constants.FO_FmtRule	int <b>Value: 59</b>
Constants.FO_FmtRuleClause	int <b>Value: 60</b>
Constants.FO_TiFlow	int <b>Value: 61</b>
Constants.FO_TiText	int <b>Value: 62</b>
Constants.FO_TiTextTable	int <b>Value: 63</b>
Constants.FO_TiApiClient	int <b>Value: 64</b>
Constants.FO_SubCol	int <b>Value: 65</b>
Constants.FO_MarkerType	int <b>Value: 66</b>
Constants.FO_CombinedFontDefn	int <b>Value: 67</b>
Constants.FO_Rubi	int <b>Value: 68</b>
Constants.FO_Alert	int <b>Value: 69</b>
Constants.FO_CursorResource	int <b>Value: 70</b>
Constants.FO_AttrCondExpr	int <b>Value: 71</b>
Constants.FO_DlgListView	int <b>Value: 72</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FO_GraphicsFmt	int <b>Value:</b> 73
Constants.FO_CMSRegistration	int <b>Value:</b> 74
Constants.FO_CMSSession	int <b>Value:</b> 75
Constants.FO_CMSObject	int <b>Value:</b> 76
Constants.FO_KeyCatalog	int <b>Value:</b> 77
Constants.FO_Iterator	int <b>Value:</b> 78
Constants.FO_Num	int <b>Value:</b> 79
Constants.FO_First_Internal	int <b>Value:</b> 79
Constants.FO_Page	int <b>Value:</b> (79+0)
Constants.FO_Graphic	int <b>Value:</b> (79+1)
Constants.FO_Frame	int <b>Value:</b> (79+2)
Constants.FO_Cblock	int <b>Value:</b> (79+3)
Constants.FO_SubCond	int <b>Value:</b> (79+4)
Constants.FO_SubStyle	int <b>Value:</b> (79+5)
Constants.FO_SubDash	int <b>Value:</b> (79+6)
Constants.FO_SubCell	int <b>Value:</b> (79+7)
Constants.FO_SubTbl	int <b>Value:</b> (79+8)
Constants.FO_SubPblock	int <b>Value:</b> (79+9)

<b>Property name</b>	<b>Data Type</b>
Constants.FO_SubTextDef	int <b>Value:</b> (79+10)
Constants.FO_MenuCell	int <b>Value:</b> (79+11)
Constants.FO_TextInset	int <b>Value:</b> (79+12)
Constants.FO_DitaMap	int <b>Value:</b> (79+13)
Constants.FO_Last_Internal	int <b>Value:</b> (79+14)
Constants.FO_CMSTConnection	int <b>Value:</b> (79+15)
Constants.FO_CMSTObject2	int <b>Value:</b> (79+16)
Constants.FO_Bad	int <b>Value:</b> 255
Constants.FO_AnchoredFrame	int <b>Value:</b> 23
Constants.FO_UFrame	int <b>Value:</b> 9
Constants.FP_Name	int <b>Value:</b> 20
Constants.FP_Unique	int <b>Value:</b> 21
Constants.FP_UserString	int <b>Value:</b> 22
Constants.FP_ObjectAttributes	int <b>Value:</b> 23
Constants.FP_Label	int <b>Value:</b> 30
Constants.FP_IsIconified	int <b>Value:</b> 31
Constants.FP_IsInFront	int <b>Value:</b> 32
Constants.FP_ScreenX	int <b>Value:</b> 33

<b>Property name</b>	<b>Data Type</b>
Constants.FP_ScreenY	int <b>Value: 34</b>
Constants.FP_ScreenWidth	int <b>Value: 35</b>
Constants.FP_ScreenHeight	int <b>Value: 36</b>
Constants.FV_SessionId	int <b>Value: 0</b>
Constants.FP_VersionRevision	int <b>Value: 41</b>
Constants.FP_ProductName	int <b>Value: 42</b>
Constants.FP_VersionMajor	int <b>Value: 43</b>
Constants.FP_VersionMinor	int <b>Value: 44</b>
Constants.FP_Platform	int <b>Value: 45</b>
Constants.FP_OperatingSystem	int <b>Value: 46</b>
Constants.FP_WindowSystem	int <b>Value: 47</b>
Constants.FP_AutoSave	int <b>Value: 48</b>
Constants.FP_AutoSaveSeconds	int <b>Value: 49</b>
Constants.FP_AutoBackup	int <b>Value: 50</b>
Constants.FP_ActiveDoc	int <b>Value: 51</b>
Constants.FP_ProgId	int <b>Value: 2421</b>
Constants.FP_IsFMRRunningAsServer	int <b>Value: 2348</b>
Constants.FP_ActiveView	int <b>Value: 2422</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_CurrentDoc	int <b>Value:</b> 2401
Constants.FP_PDFPageNum	int <b>Value:</b> 2402
Constants.FP_UndoFDKRecording	int <b>Value:</b> 2403
Constants.FP_StackWarningLevel	int <b>Value:</b> 2404
Constants.FP_NoFlashInPDF	int <b>Value:</b> 2405
Constants.FP_No3DInPDF	int <b>Value:</b> 2406
Constants.FP_InsetURL	int <b>Value:</b> 2407
Constants.FP_DontShowWelcomeScreen	int <b>Value:</b> 2408
Constants.FP_TechSuiteInternal	int <b>Value:</b> 2409
Constants.FP_IsTechnicalSuiteLicensed	int <b>Value:</b> 2410
Constants.FP_TechSuiteInternal2	int <b>Value:</b> 2411
Constants.FP_IsFMRunningInTrialPeriod	int <b>Value:</b> 2412
Constants.FP_EnableAutoSpellCheck	int <b>Value:</b> 2413
Constants.FP_CustomDelimiter	int <b>Value:</b> 2420
Constants.FV_WarnNever	int <b>Value:</b> 0
Constants.FV_WarnAlways	int <b>Value:</b> 1
Constants.FV_WarnOnce	int <b>Value:</b> 2
Constants.FV_UndoAll	int <b>Value:</b> 3

<b>Property name</b>	<b>Data Type</b>
Constants.FV_RedoAll	int <b>Value:</b> 4
Constants.FP_ActiveBook	int <b>Value:</b> 52
Constants.FP_FirstOpenDoc	int <b>Value:</b> 53
Constants.FP_FirstOpenBook	int <b>Value:</b> 54
Constants.FP_FontFamilyNames	int <b>Value:</b> 55
Constants.FP_FontVariationNames	int <b>Value:</b> 56
Constants.FP_FontWeightNames	int <b>Value:</b> 57
Constants.FP_FontAngleNames	int <b>Value:</b> 58
Constants.FP_MarkerNames	int <b>Value:</b> 59
Constants.FP_FontFamilyAttributes	int <b>Value:</b> 60
Constants.FV_FAMILY_VISIBLE	int <b>Value:</b> 0x00000001
Constants.FV_FAMILY_SELECTABLE	int <b>Value:</b> 0x00000002
Constants.FV_FAMILY_MAPPED	int <b>Value:</b> 0x00000004
Constants.FV_FAMILY_SURROGATE	int <b>Value:</b> 0xFFFF0000
Constants.FP_RememberMissingFontNames	int <b>Value:</b> 61
Constants.FP_Mif8bitOutput	int <b>Value:</b> 62
Constants.FP_GreekSize	int <b>Value:</b> 63
Constants.FP_RetainUndoState	int <b>Value:</b> 64



<b>Property name</b>	<b>Data Type</b>
Constants.FP_ProductIsStructured	int <b>Value: 65</b>
Constants.FP_ProductIsDemo	int <b>Value: 66</b>
Constants.FP_DisableAutofitAfterZoom	int <b>Value: 67</b>
Constants.FP_DefaultFontFamily	int <b>Value: 68</b>
Constants.FP_DefaultFontVariation	int <b>Value: 69</b>
Constants.FP_DefaultFontWeight	int <b>Value: 70</b>
Constants.FP_DefaultFontAngle	int <b>Value: 71</b>
Constants.FP_CTFontContext	int <b>Value: 72</b>
Constants.FP_FontFamilyFullNames	int <b>Value: 73</b>
Constants.FP_RpcPropertyName	int <b>Value: 76</b>
Constants.FP_RpcProgramNumber	int <b>Value: 77</b>
Constants.FP_DisplayName	int <b>Value: 78</b>
Constants.FP_ProcessNumber	int <b>Value: 79</b>
Constants.FP_OpenDir	int <b>Value: 80</b>
Constants.FP_HostName	int <b>Value: 81</b>
Constants.FP_UserName	int <b>Value: 82</b>
Constants.FP_UserLogin	int <b>Value: 83</b>
Constants.FP_UserHomeDir	int <b>Value: 84</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_UserSettingsDir	int <b>Value:</b> 152
Constants.FP_Path	int <b>Value:</b> 85
Constants.FP_TmpDir	int <b>Value:</b> 86
Constants.FP_FM_HomeDir	int <b>Value:</b> 87
Constants.FP_FM_BinDir	int <b>Value:</b> 88
Constants.FP_FM_InitDir	int <b>Value:</b> 89
Constants.FP_FM_CurrentDir	int <b>Value:</b> 90
Constants.FP_FM_SgmlDir	int <b>Value:</b> 100
Constants.FP_FM_StructureDir	int <b>Value:</b> 2031
Constants.FP_FM_XmlDir	int <b>Value:</b> 2032
Constants.FP_Reformatting	int <b>Value:</b> 91
Constants.FP_Displaying	int <b>Value:</b> 92
Constants.FP_ApplyFormatRules	int <b>Value:</b> 93
Constants.FP_ApplyEOPRules	int <b>Value:</b> 2030
Constants.FP_Validating	int <b>Value:</b> 94
Constants.FP_ToolBar	int <b>Value:</b> 95
Constants.FP_FirstMenuItemInSession	int <b>Value:</b> 97
Constants.FP_FirstCommandInSession	int <b>Value:</b> 98

<b>Property name</b>	<b>Data Type</b>
Constants.FP_CurrentMenuSet	int <b>Value: 99</b>
Constants.FV_MENU_QUICK	int <b>Value: 1</b>
Constants.FV_MENU_COMPLETE	int <b>Value: 2</b>
Constants.FV_MENU_CUSTOM	int <b>Value: 3</b>
Constants.FP_IconBarOn	int <b>Value: 101</b>
Constants.FP_HelpPending	int <b>Value: 102</b>
Constants.FP_FM_HelpDir	int <b>Value: 103</b>
Constants.FP_PercentDone	int <b>Value: 104</b>
Constants.FP_ActiveAlert	int <b>Value: 105</b>
Constants.FP_Snap	int <b>Value: 151</b>
Constants.FP_Gravity	int <b>Value: 107</b>
Constants.FP_KByteAllocationSize	int <b>Value: 106</b>
Constants.FP_ImportFilters	int <b>Value: 2288</b>
Constants.FP_ExportFilters	int <b>Value: 2289</b>
Constants.FP_UIColorBackground	int <b>Value: 2297</b>
Constants.FP_UIColorRuler	int <b>Value: 2298</b>
Constants.FP_BIBGetAddressProc	int <b>Value: 2299</b>
Constants.FP_RasterFilterDpi	int <b>Value: 2400</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_Hint_HintVersion	int <b>Value: 0</b>
Constants.FV_Hint_VendorId	int <b>Value: 1</b>
Constants.FV_Hint_FormatId	int <b>Value: 2</b>
Constants.FV_Hint_Platform	int <b>Value: 3</b>
Constants.FV_Hint_FilterVersion	int <b>Value: 4</b>
Constants.FV_Hint_FilterName	int <b>Value: 5</b>
Constants.FP_DefaultKeyCatalog	int <b>Value: 2871</b>
Constants.FP_FirstKeyCatalogInSession	int <b>Value: 2872</b>
Constants.FP_KeyCatalogWorkflow	int <b>Value: 2859</b>
Constants.FV_KeyCatalogWorkflowAuthoring	int <b>Value: 0</b>
Constants.FV_KeyCatalogWorkflowPublishing	int <b>Value: 1</b>
Constants.FV_KeyCatalogWorkflowSearching	int <b>Value: 2</b>
Constants.FV_KeyCatalogWorkflowMax	int <b>Value: 3</b>
Constants.FP_StatusLine	int <b>Value: 108</b>
Constants.FP_TextSelection	int <b>Value: 109</b>
Constants.FP_IsOnScreen	int <b>Value: 110</b>
Constants.FP_NextOpenDocInSession	int <b>Value: 111</b>
Constants.FP_FirstGraphicInDoc	int <b>Value: 112</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_FirstPgflnDoc	int <b>Value:</b> 113
Constants.FP_FirstMarkerInDoc	int <b>Value:</b> 114
Constants.FP_FirstVarInDoc	int <b>Value:</b> 115
Constants.FP_FirstVarFmtInDoc	int <b>Value:</b> 116
Constants.FP_FirstXRefInDoc	int <b>Value:</b> 117
Constants.FP_FirstXRefFmtInDoc	int <b>Value:</b> 118
Constants.FP_FirstFnInDoc	int <b>Value:</b> 119
Constants.FP_FirstTblInDoc	int <b>Value:</b> 120
Constants.FP_FirstFlowInDoc	int <b>Value:</b> 121
Constants.FP_FirstPgffmtInDoc	int <b>Value:</b> 122
Constants.FP_FirstCharFmtInDoc	int <b>Value:</b> 123
Constants.FP_FirstCondFmtInDoc	int <b>Value:</b> 124
Constants.FP_FirstTblFmtInDoc	int <b>Value:</b> 125
Constants.FP_FirstRulingFmtInDoc	int <b>Value:</b> 126
Constants.FP_FirstColorInDoc	int <b>Value:</b> 127
Constants.FP_FirstSelectedGraphicInDoc	int <b>Value:</b> 128
Constants.FP_SelectedTbl	int <b>Value:</b> 129
Constants.FP_FileExtensionOverride	int <b>Value:</b> 2813

<b>Property name</b>	<b>Data Type</b>
Constants.FP_FirstAttrCondExprInDoc	int <b>Value:</b> 2815
Constants.FP_DocOpenType	int <b>Value:</b> 130
Constants.FV_DOC_TYPE_BINARY	int <b>Value:</b> 0x01
Constants.FV_DOC_TYPE_TEXT	int <b>Value:</b> 0x02
Constants.FV_DOC_TYPE_MIF	int <b>Value:</b> 0x03
Constants.FV_DOC_TYPE_FILTER	int <b>Value:</b> 0x04
Constants.FP_DocSaveFormat	int <b>Value:</b> 2415
Constants.FP_DocIsModified	int <b>Value:</b> 131
Constants.FP_DocIsHelp	int <b>Value:</b> 132
Constants.FP_DocIsViewOnly	int <b>Value:</b> 133
Constants.FP_ViewOnlyWinPalette	int <b>Value:</b> 134
Constants.FP_ViewOnlyWinMenubar	int <b>Value:</b> 135
Constants.FP_ViewOnlyWinBorders	int <b>Value:</b> 136
Constants.FP_ViewOnlyWinPopup	int <b>Value:</b> 137
Constants.FP_ViewOnlyXRef	int <b>Value:</b> 138
Constants.FV_VOX_NOT_ACTIVE	int <b>Value:</b> 0
Constants.FV_VOX_GOTO_BEHAVIOR	int <b>Value:</b> 1
Constants.FV_VOX_OPEN_BEHAVIOR	int <b>Value:</b> 2

<b>Property name</b>	<b>Data Type</b>
Constants.FV_VOX_ALERT	int <b>Value: 3</b>
Constants.FP_ViewOnlySelect	int <b>Value: 139</b>
Constants.FV_VOS_USER_ONLY	int <b>Value: 1</b>
Constants.FV_VOS_NONE	int <b>Value: 2</b>
Constants.FV_VOS_YES	int <b>Value: 3</b>
Constants.FP_ViewOnlyDeadCodes	int <b>Value: 140</b>
Constants.FP_FirstTilnDoc	int <b>Value: 141</b>
Constants.FP_FirstSelectedTilnDoc	int <b>Value: 142</b>
Constants.FP_DocIsViewOnlyWinPalette	int <b>Value: 134</b>
Constants.FP_MenuBar	int <b>Value: 143</b>
Constants.FP_ViewOnlyMenuBar	int <b>Value: 144</b>
Constants.FP_DocSaveType	int <b>Value: 145</b>
Constants.FP_Untouchable	int <b>Value: 146</b>
Constants.FP_DocFluidFlow	int <b>Value: 147</b>
Constants.FP_FirstMarkerTypeInDoc	int <b>Value: 148</b>
Constants.FP_MarkerTypeNames	int <b>Value: 149</b>
Constants.FP_AddMarkerTypeToStandardMarkers	int <b>Value: 150</b>
Constants.FP_ShowAll	int <b>Value: 155</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_ShowCondIndicators	int <b>Value:</b> 156
Constants.FP_DontUpdateXRefs	int <b>Value:</b> 159
Constants.FP_DontUpdateTextInsets	int <b>Value:</b> 160
Constants.FP_LineBreakAfter	int <b>Value:</b> 177
Constants.FP_SuperScriptSize	int <b>Value:</b> 178
Constants.FP_SuperScriptShift	int <b>Value:</b> 179
Constants.FP_SubScriptSize	int <b>Value:</b> 180
Constants.FP_SubScriptShift	int <b>Value:</b> 181
Constants.FP_SmallCapsSize	int <b>Value:</b> 182
Constants.FP_SuperScriptStretch	int <b>Value:</b> 183
Constants.FP_SubScriptStretch	int <b>Value:</b> 184
Constants.FP_SmallCapsStretch	int <b>Value:</b> 185
Constants.FP_RubiSize	int <b>Value:</b> 186
Constants.FP_RubiFixedSize	int <b>Value:</b> 187
Constants.FP_NarrowRubiSpaceForJapanese	int <b>Value:</b> 188
Constants.FP_WideRubiSpaceForJapanese	int <b>Value:</b> 189
Constants.FP_NarrowRubiSpaceForOther	int <b>Value:</b> 190
Constants.FP_WideRubiSpaceForOther	int <b>Value:</b> 191



<b>Property name</b>	<b>Data Type</b>
Constants.FV_WIDE	int <b>Value: 0</b>
Constants.FV_NARROW	int <b>Value: 1</b>
Constants.FV_PROPORTIONAL	int <b>Value: 2</b>
Constants.FP_RubiOverhang	int <b>Value: 192</b>
Constants.FP_RubiAlignAtLineBounds	int <b>Value: 193</b>
Constants.FP_FirstRubiInDoc	int <b>Value: 194</b>
Constants.FP_ScreenCaptureDocToFile	int <b>Value: 195</b>
Constants.FP_Dictionary	int <b>Value: 203</b>
Constants.FP_VolNumComputeMethod	int <b>Value: 211</b>
Constants.FV_NUM_READ_FROM_FILE	int <b>Value: 0x00</b>
Constants.FV_NUM_CONTINUE	int <b>Value: 0x01</b>
Constants.FV_NUM_RESTART	int <b>Value: 0x02</b>
Constants.FV_NUM_SAME	int <b>Value: 0x03</b>
Constants.FV_NUM_PERPAGE	int <b>Value: 0x03</b>
Constants.FP_VolumeNumber	int <b>Value: 212</b>
Constants.FP_VolumeNumStyle	int <b>Value: 213</b>
Constants.FP_VolumeNumText	int <b>Value: 214</b>
Constants.FP_ChapNumComputeMethod	int <b>Value: 215</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_ChapterNumber	int <b>Value:</b> 216
Constants.FP_ChapterNumStyle	int <b>Value:</b> 217
Constants.FV_NUMSTYLE_NUMERIC	int <b>Value:</b> 0x00
Constants.FV_NUMSTYLE_ROMAN_UC	int <b>Value:</b> 0x01
Constants.FV_NUMSTYLE_ROMAN_LC	int <b>Value:</b> 0x02
Constants.FV_NUMSTYLE_ALPHA_UC	int <b>Value:</b> 0x03
Constants.FV_NUMSTYLE_ALPHA_LC	int <b>Value:</b> 0x04
Constants.FV_NUMSTYLE_KANJI	int <b>Value:</b> 0x05
Constants.FV_NUMSTYLE_ZENKAKU	int <b>Value:</b> 0x06
Constants.FV_NUMSTYLE_ZENKAKU_UC	int <b>Value:</b> 0x07
Constants.FV_NUMSTYLE_ZENKAKU_LC	int <b>Value:</b> 0x08
Constants.FV_NUMSTYLE_KANJI_KAZU	int <b>Value:</b> 0x09
Constants.FV_NUMSTYLE_DAIJI	int <b>Value:</b> 0x0a
Constants.FV_NUMSTYLE_TEXT	int <b>Value:</b> 0x0b
Constants.FV_NUMSTYLE_FULLWIDTH	int <b>Value:</b> 0x0c
Constants.FV_NUMSTYLE_FULLWIDTH_UC	int <b>Value:</b> 0x0d
Constants.FV_NUMSTYLE_FULLWIDTH_LC	int <b>Value:</b> 0x0e
Constants.FV_NUMSTYLE_CHINESE_NUMERIC	int <b>Value:</b> 0x10

**Object Reference**

<b>Property name</b>	<b>Data Type</b>
Constants.FP_ChapterNumText	int <b>Value:</b> 218
Constants.FP_FirstPageNum	int <b>Value:</b> 224
Constants.FP_PageNumStyle	int <b>Value:</b> 225
Constants.FV_PAGE_NUM_NUMERIC	int <b>Value:</b> 0x00
Constants.FV_PAGE_NUM_ROMAN_UC	int <b>Value:</b> 0x01
Constants.FV_PAGE_NUM_ROMAN_LC	int <b>Value:</b> 0x02
Constants.FV_PAGE_NUM_ALPHA_UC	int <b>Value:</b> 0x03
Constants.FV_PAGE_NUM_ALPHA_LC	int <b>Value:</b> 0x04
Constants.FV_PAGE_NUM_KANJI	int <b>Value:</b> 0x05
Constants.FV_PAGE_NUM_ZENKAKU	int <b>Value:</b> 0x06
Constants.FV_PAGE_NUM_ZENKAKU_UC	int <b>Value:</b> 0x07
Constants.FV_PAGE_NUM_ZENKAKU_LC	int <b>Value:</b> 0x08
Constants.FV_PAGE_NUM_KANJI_KAZU	int <b>Value:</b> 0x09
Constants.FV_PAGE_NUM_DAIJI	int <b>Value:</b> 0x0a
Constants.FV_PAGE_NUM_FULLWIDTH	int <b>Value:</b> 0x0c
Constants.FV_PAGE_NUM_FULLWIDTH_UC	int <b>Value:</b> 0x0d
Constants.FV_PAGE_NUM_FULLWIDTH_LC	int <b>Value:</b> 0x0e
Constants.FV_PAGE_NUM_CHINESE_NUMERIC	int <b>Value:</b> 0x10

## Object Reference

Property name	Data Type
Constants.FP_DocIsDoubleSided	int <b>Value:</b> 226
Constants.FP_FirstPageVerso	int <b>Value:</b> 227
Constants.FP_PointPageNumStyle	int <b>Value:</b> 228
Constants.FV_POINT_PAGE_NUM_NUMERIC	int <b>Value:</b> 0x00
Constants.FV_POINT_PAGE_NUM_ROMAN_UC	int <b>Value:</b> 0x01
Constants.FV_POINT_PAGE_NUM_ROMAN_LC	int <b>Value:</b> 0x02
Constants.FV_POINT_PAGE_NUM_ALPHA_UC	int <b>Value:</b> 0x03
Constants.FV_POINT_PAGE_NUM_ALPHA_LC	int <b>Value:</b> 0x04
Constants.FV_POINT_PAGE_NUM_KANJI	int <b>Value:</b> 0x05
Constants.FV_POINT_PAGE_NUM_ZENKAKU	int <b>Value:</b> 0x06
Constants.FV_POINT_PAGE_NUM_ZENKAKU_UC	int <b>Value:</b> 0x07
Constants.FV_POINT_PAGE_NUM_ZENKAKU_LC	int <b>Value:</b> 0x08
Constants.FV_POINT_PAGE_NUM_KANJI_KAZU	int <b>Value:</b> 0x09
Constants.FV_POINT_PAGE_NUM_DAIJI	int <b>Value:</b> 0x0a
Constants.FV_POINT_PAGE_FULLWIDTH	int <b>Value:</b> 0x0c
Constants.FV_POINT_PAGE_FULLWIDTH_UC	int <b>Value:</b> 0x0d
Constants.FV_POINT_PAGE_FULLWIDTH_LC	int <b>Value:</b> 0x0e
Constants.FV_POINT_PAGE_CHINESE_NUMERIC	int <b>Value:</b> 0x10

<b>Property name</b>	<b>Data Type</b>
Constants.FP_PageRounding	int <b>Value:</b> 229
Constants.FV_PR_DEL_EMPTY	int <b>Value:</b> 0x01
Constants.FV_PR_KEEP_NUM_EVEN	int <b>Value:</b> 0x02
Constants.FV_PR_KEEP_NUM_ODD	int <b>Value:</b> 0x03
Constants.FV_PR_DONT_CHANGE	int <b>Value:</b> 0x04
Constants.FP_TopMargin	int <b>Value:</b> 230
Constants.FP_BottomMargin	int <b>Value:</b> 231
Constants.FP_LeftMargin	int <b>Value:</b> 232
Constants.FP_RightMargin	int <b>Value:</b> 233
Constants.FP_ColGap	int <b>Value:</b> 234
Constants.FP_NumCols	int <b>Value:</b> 235
Constants.FP_CurrentPage	int <b>Value:</b> 236
Constants.FP_SmartQuotes	int <b>Value:</b> 237
Constants.FP_AutoChangeBars	int <b>Value:</b> 238
Constants.FP_SmartSpaces	int <b>Value:</b> 239
Constants.FP_CurrentInset	int <b>Value:</b> 240
Constants.FP_ChangeBarDistance	int <b>Value:</b> 259
Constants.FP_ChangeBarPosition	int <b>Value:</b> 260

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

<b>Property name</b>	<b>Data Type</b>
Constants.FV_FN_NUM_ROMAN_UC	int <b>Value:</b> 0x01
Constants.FV_FN_NUM_ROMAN_LC	int <b>Value:</b> 0x02
Constants.FV_FN_NUM_ALPHA_UC	int <b>Value:</b> 0x03
Constants.FV_FN_NUM_ALPHA_LC	int <b>Value:</b> 0x04
Constants.FV_FN_NUM_CUSTOM	int <b>Value:</b> 0x05
Constants.FV_FN_NUM_KANJI	int <b>Value:</b> 0x06
Constants.FV_FN_NUM_ZENKAKU	int <b>Value:</b> 0x07
Constants.FV_FN_NUM_ZENKAKU_UC	int <b>Value:</b> 0x08
Constants.FV_FN_NUM_ZENKAKU_LC	int <b>Value:</b> 0x09
Constants.FV_FN_NUM_KANJI_KAZU	int <b>Value:</b> 0x0A
Constants.FV_FN_NUM_DAIJI	int <b>Value:</b> 0x0B
Constants.FV_FN_NUM_FULL_WIDTH	int <b>Value:</b> 0x0C
Constants.FV_FN_NUM_FULL_WIDTH_UC	int <b>Value:</b> 0x0D
Constants.FV_FN_NUM_FULL_WIDTH_LC	int <b>Value:</b> 0x0E
Constants.FV_FN_NUM_CHINESE_NUMERIC	int <b>Value:</b> 0x10
Constants.FP_FnNumberingPerPage	int <b>Value:</b> 291
Constants.FP_FnRefPosition	int <b>Value:</b> 292
Constants.FP_FnRefSuffix	int <b>Value:</b> 293

<b>Property name</b>	<b>Data Type</b>
Constants.FP_FnInstancePrefix	int <b>Value: 294</b>
Constants.FP_FnNumComputeMethod	int <b>Value: 295</b>
Constants.FP_TblFnFmt	int <b>Value: 315</b>
Constants.FP_TblFnNumStyle	int <b>Value: 316</b>
Constants.FP_TblFnCustNumString	int <b>Value: 317</b>
Constants.FP_TblFnCellPosition	int <b>Value: 318</b>
Constants.FP_TblFnCellSuffix	int <b>Value: 319</b>
Constants.FP_TblFnCellPrefix	int <b>Value: 320</b>
Constants.FP_TblFnPosition	int <b>Value: 321</b>
Constants.FP_TblFnPrefix	int <b>Value: 322</b>
Constants.FP_TblFnSuffix	int <b>Value: 323</b>
Constants.FP_TblFnNumComputeMethod	int <b>Value: 324</b>
Constants.FP_Symbols	int <b>Value: 334</b>
Constants.FP_SymbolsList	int <b>Value: 335</b>
Constants.FP_Variables	int <b>Value: 336</b>
Constants.FP_Strings	int <b>Value: 337</b>
Constants.FP_Numbers	int <b>Value: 338</b>
Constants.FP_Functions	int <b>Value: 339</b>



<b>Property name</b>	<b>Data Type</b>
Constants.FP_HorizontalSpreadSmall	int <b>Value:</b> 340
Constants.FP_HorizontalSpreadMed	int <b>Value:</b> 341
Constants.FP_HorizontalSpreadLarge	int <b>Value:</b> 342
Constants.FP_VerticalSpreadSmall	int <b>Value:</b> 343
Constants.FP_VerticalSpreadMed	int <b>Value:</b> 344
Constants.FP_VerticalSpreadLarge	int <b>Value:</b> 345
Constants.FP_EqnIntegralSizeSmall	int <b>Value:</b> 346
Constants.FP_EqnIntegralSizeMed	int <b>Value:</b> 347
Constants.FP_EqnIntegralSizeLarge	int <b>Value:</b> 348
Constants.FP_EqnSigmaSizeSmall	int <b>Value:</b> 349
Constants.FP_EqnSigmaSizeMed	int <b>Value:</b> 350
Constants.FP_EqnSigmaSizeLarge	int <b>Value:</b> 351
Constants.FP_EqnLevel1SizeSmall	int <b>Value:</b> 352
Constants.FP_EqnLevel1SizeMed	int <b>Value:</b> 353
Constants.FP_EqnLevel1SizeLarge	int <b>Value:</b> 354
Constants.FP_EqnLevel2SizeSmall	int <b>Value:</b> 355
Constants.FP_EqnLevel2SizeMed	int <b>Value:</b> 356
Constants.FP_EqnLevel2SizeLarge	int <b>Value:</b> 357

<b>Property name</b>	<b>Data Type</b>
Constants.FP_EqnLevel3SizeSmall	int <b>Value:</b> 358
Constants.FP_EqnLevel3SizeMed	int <b>Value:</b> 359
Constants.FP_EqnLevel3SizeLarge	int <b>Value:</b> 360
Constants.FP_ViewBorders	int <b>Value:</b> 382
Constants.FP_ViewHotspotIndicators	int <b>Value:</b> 2357
Constants.FP_ViewRulers	int <b>Value:</b> 383
Constants.FP_ViewNoGraphics	int <b>Value:</b> 384
Constants.FP_ViewPageScrolling	int <b>Value:</b> 385
Constants.FV_SCROLL_VARIABLE	int <b>Value:</b> 0
Constants.FV_SCROLL_HORIZONTAL	int <b>Value:</b> 1
Constants.FV_SCROLL_VERTICAL	int <b>Value:</b> 2
Constants.FV_SCROLL_FACING	int <b>Value:</b> 3
Constants.FP_ViewGridUnits	int <b>Value:</b> 386
Constants.FP_Zoom	int <b>Value:</b> 387
Constants.FV_NO_ZOOM_MANGLING	int <b>Value:</b> -1
Constants.FV_ZOOM_TO_FIT_TO_WINDOW	int <b>Value:</b> -2
Constants.FV_FIT_WINDOW_TO_PAGE	int <b>Value:</b> -3
Constants.FP_ViewTextSymbols	int <b>Value:</b> 388

<b>Property name</b>	<b>Data Type</b>
Constants.FP_ViewGrid	int <b>Value:</b> 389
Constants.FP_ViewDisplayUnits	int <b>Value:</b> 390
Constants.FP_TrackChangesOn	int <b>Value:</b> 2819
Constants.FP_PreviewState	int <b>Value:</b> 2820
Constants.FV_PREVIEW_OFF_TRACK_CHANGE	int <b>Value:</b> 0
Constants.FV_PREVIEW_ON_ORIGINAL	int <b>Value:</b> 1
Constants.FV_PREVIEW_ON_FINAL	int <b>Value:</b> 2
Constants.FP_ReviewerNameList	int <b>Value:</b> 2825
Constants.FP_TrackChangesAddedColor	int <b>Value:</b> 2826
Constants.FP_TrackChangesDeletedColor	int <b>Value:</b> 2827
Constants.FP_ViewRulerUnits	int <b>Value:</b> 391
Constants.FP_SpotColorView	int <b>Value:</b> 392
Constants.FP_SnapGridUnits	int <b>Value:</b> 393
Constants.FP_SnapAngle	int <b>Value:</b> 394
Constants.FP_ViewLinkBoundaries	int <b>Value:</b> 395
Constants.FP_ViewFontSizeUnits	int <b>Value:</b> 396
Constants.FP_LeftMasterPage	int <b>Value:</b> 413
Constants.FP_RightMasterPage	int <b>Value:</b> 414

<b>Property name</b>	<b>Data Type</b>
Constants.FP_FirstBodyPageInDoc	int <b>Value:</b> 415
Constants.FP_LastBodyPageInDoc	int <b>Value:</b> 416
Constants.FP_FirstMasterPageInDoc	int <b>Value:</b> 417
Constants.FP_LastMasterPageInDoc	int <b>Value:</b> 418
Constants.FP_FirstRefPageInDoc	int <b>Value:</b> 419
Constants.FP_LastRefPageInDoc	int <b>Value:</b> 420
Constants.FP_HiddenPage	int <b>Value:</b> 421
Constants.FP_MainFlowInDoc	int <b>Value:</b> 422
Constants.FP_PrintStartPageName	int <b>Value:</b> 439
Constants.FP_PrintEndPageName	int <b>Value:</b> 440
Constants.FP_PrintCols	int <b>Value:</b> 441
Constants.FP_PrintRows	int <b>Value:</b> 442
Constants.FP_PrintStartPage	int <b>Value:</b> 443
Constants.FP_PrintStartPoint	int <b>Value:</b> 444
Constants.FP_PrintOddPages	int <b>Value:</b> 445
Constants.FP_PrintCollated	int <b>Value:</b> 446
Constants.FP_PrintLowRes	int <b>Value:</b> 447
Constants.FP_PrintThumbnails	int <b>Value:</b> 448

<b>Property name</b>	<b>Data Type</b>
Constants.FP_PrinterName	int <b>Value:</b> 449
Constants.FP_PrintScale	int <b>Value:</b> 450
Constants.FP_PrintFileName	int <b>Value:</b> 451
Constants.FP_PrintDitavalFileName	int <b>Value:</b> 2343
Constants.FP_PrintScope	int <b>Value:</b> 452
Constants.FV_PR_ALL	int <b>Value:</b> 1
Constants.FV_PR_RANGE	int <b>Value:</b> 2
Constants.FP_PrintEndPage	int <b>Value:</b> 453
Constants.FP_PrintEndPoint	int <b>Value:</b> 454
Constants.FP_PrintEvenPages	int <b>Value:</b> 455
Constants.FP_PrintLastSheetFirst	int <b>Value:</b> 456
Constants.FP_PrintRegistrationMarks	int <b>Value:</b> 457
Constants.FP_PrintManualFeed	int <b>Value:</b> 458
Constants.FP_PrintNumCopies	int <b>Value:</b> 459
Constants.FP_PrintToFile	int <b>Value:</b> 460
Constants.FP_PrintPaperWidth	int <b>Value:</b> 461
Constants.FP_PrintPaperHeight	int <b>Value:</b> 462
Constants.FP_PrintSepts	int <b>Value:</b> 463

<b>Property name</b>	<b>Data Type</b>
Constants.FP_SkipBlankSepts	int <b>Value: 464</b>
Constants.FP_PrintImaging	int <b>Value: 465</b>
Constants.FV_IMG_POSITIVE	int <b>Value: 0</b>
Constants.FV_IMG_NEGATIVE	int <b>Value: 1</b>
Constants.FP_PrintEmulsion	int <b>Value: 466</b>
Constants.FV_EMUL_UP	int <b>Value: 0</b>
Constants.FV_EMUL_DOWN	int <b>Value: 1</b>
Constants.FP_PrintBlankPages	int <b>Value: 467</b>
Constants.FP_PrintTomboMarks	int <b>Value: 468</b>
Constants.FP_PrintRegMarkDate	int <b>Value: 469</b>
Constants.FP_TrapwiseCompatibility	int <b>Value: 495</b>
Constants.FP_DownloadFonts	int <b>Value: 496</b>
Constants.FP_PrintSpotBW	int <b>Value: 498</b>
Constants.FV_PR_DOWNLOAD_NONE	int <b>Value: 1</b>
Constants.FV_PR_DOWNLOAD_ALL	int <b>Value: 2</b>
Constants.FV_PR_DOWNLOAD_ALL_BUT_STANDARD_13	int <b>Value: 3</b>
Constants.FV_PR_DOWNLOAD_ALL_BUT_STANDARD_35	int <b>Value: 4</b>
Constants.FP_PrintDownloadAsianFonts	int <b>Value: 501</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_PrintDownloadTrueTypeAsType1	int <b>Value:</b> 502
Constants.FP_GenerateAcrobatInfo	int <b>Value:</b> 2250
Constants.FP_AcrobatBookmarkDisplayTags	int <b>Value:</b> 2251
Constants.FP_DocAcrobatDefaultsChanged	int <b>Value:</b> 2252
Constants.FP_DocAcrobatElements	int <b>Value:</b> 2253
Constants.FP_DocAcrobatElementList	int <b>Value:</b> 2254
Constants.FP_DocAcrobatNoArticleThreads	int <b>Value:</b> 2255
Constants.FP_DocAcrobatColumnArticleThreads	int <b>Value:</b> 2256
Constants.FP_PDFAIINamedDestinations	int <b>Value:</b> 2257
Constants.FP_PDFDestsMarked	int <b>Value:</b> 2258
Constants.FP_PDFStructure	int <b>Value:</b> 2259
Constants.FP_PDFDocInfo	int <b>Value:</b> 2260
Constants.FP_PDFBookmark	int <b>Value:</b> 2261
Constants.FP_FileInfoPacket	int <b>Value:</b> 2262
Constants.FP_PDFJobOption	int <b>Value:</b> 2263
Constants.FP_PDFOpenPage	int <b>Value:</b> 2264
Constants.FP_PDFZoomType	int <b>Value:</b> 2265
Constants.FV_PDFZoomNone	int <b>Value:</b> 0

<b>Property name</b>	<b>Data Type</b>
Constants.FV_PDFZoomDefault	int <b>Value: 1</b>
Constants.FV_PDFZoomPage	int <b>Value: 2</b>
Constants.FV_PDFZoomWidth	int <b>Value: 3</b>
Constants.FV_PDFZoomHeight	int <b>Value: 4</b>
Constants.FV_PDFZoomMaxValue	int <b>Value: 4</b>
Constants.FV_PDFRegistrationMarksNone	int <b>Value: 0</b>
Constants.FV_PDFRegistrationMarksWestern	int <b>Value: 1</b>
Constants.FV_PDFRegistrationMarksTombo	int <b>Value: 2</b>
Constants.FV_PDFRegistrationMarksMax	int <b>Value: 2</b>
Constants.FP_PDFZoomFactor	int <b>Value: 2266</b>
Constants.FP_PDFSeparateFiles	int <b>Value: 2267</b>
Constants.FP_PDFRegistrationMarks	int <b>Value: 2268</b>
Constants.FP_PDFPageWidth	int <b>Value: 2269</b>
Constants.FP_PDFPageHeight	int <b>Value: 2270</b>
Constants.FP_PDFPrintPageRange	int <b>Value: 2271</b>
Constants.FP_PDFStartPage	int <b>Value: 2272</b>
Constants.FP_PDFEndPage	int <b>Value: 2273</b>
Constants.FP_PDFConvertCMYKtoRGB	int <b>Value: 2274</b>



## Object Reference

Property name	Data Type
Constants.FP_PDFBookmarksOpenLevel	int <b>Value:</b> 2275
Constants.FP_PDFDistillerAbsent	int <b>Value:</b> 2276
Constants.FP_PDFJobOptionsAbsent	int <b>Value:</b> 2277
Constants.FP_PDFViewPDF	int <b>Value:</b> 2278
Constants.FP_PDFGenerateForReview	int <b>Value:</b> 2279
Constants.FV_PDFBookmarksOpenDefaultLevel	int <b>Value:</b> -1
Constants.FV_PDFBookmarksOpenAllLevels	int <b>Value:</b> -2
Constants.FV_PDFBookmarksOpenNoneLevel	int <b>Value:</b> -3
Constants.FP_MaxFirstIndent	int <b>Value:</b> 472
Constants.FP_MinFirstIndent	int <b>Value:</b> 473
Constants.FP_MaxLeftIndent	int <b>Value:</b> 474
Constants.FP_MinLeftIndent	int <b>Value:</b> 475
Constants.FP_MaxRightIndent	int <b>Value:</b> 476
Constants.FP_MinRightIndent	int <b>Value:</b> 477
Constants.FP_MaxSpaceAbove	int <b>Value:</b> 478
Constants.FP_MinSpaceAbove	int <b>Value:</b> 479
Constants.FP_MaxSpaceBelow	int <b>Value:</b> 480
Constants.FP_MinSpaceBelow	int <b>Value:</b> 481

<b>Property name</b>	<b>Data Type</b>
Constants.FP_MaxLeading	int <b>Value: 482</b>
Constants.FP_MinLeading	int <b>Value: 483</b>
Constants.FP_MaxFontSize	int <b>Value: 484</b>
Constants.FP_MinFontSize	int <b>Value: 485</b>
Constants.FP_MaxSpread	int <b>Value: 486</b>
Constants.FP_MinSpread	int <b>Value: 487</b>
Constants.FP_MaxTabPosition	int <b>Value: 701</b>
Constants.FP_MinTabPosition	int <b>Value: 702</b>
Constants.FP_MaxLeftMargin	int <b>Value: 703</b>
Constants.FP_MinLeftMargin	int <b>Value: 704</b>
Constants.FP_MaxRightMargin	int <b>Value: 705</b>
Constants.FP_MinRightMargin	int <b>Value: 706</b>
Constants.FP_MaxTopMargin	int <b>Value: 707</b>
Constants.FP_MinTopMargin	int <b>Value: 708</b>
Constants.FP_MaxBottomMargin	int <b>Value: 709</b>
Constants.FP_MinBottomMargin	int <b>Value: 710</b>
Constants.FP_MaxStretch	int <b>Value: 711</b>
Constants.FP_MinStretch	int <b>Value: 712</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_MagicMarker	int <b>Value:</b> 488
Constants.FP_XmlVersion	int <b>Value:</b> 2800
Constants.FP_XmlEncoding	int <b>Value:</b> 2801
Constants.FP_XmlStandAlone	int <b>Value:</b> 2802
Constants.FV_XML_STANDALONE_YES	int <b>Value:</b> 1
Constants.FV_XML_STANDALONE_NO	int <b>Value:</b> 2
Constants.FV_XML_STANDALONE_NONE	int <b>Value:</b> 3
Constants.FV_XML_STANDALONE_NODEC	int <b>Value:</b> 4
Constants.FP_XmlStyleSheet	int <b>Value:</b> 2803
Constants.FP_XmlStyleSheetList	int <b>Value:</b> 2804
Constants.FP_XmlUseBOM	int <b>Value:</b> 2805
Constants.FV_XML_USEBOM_YES	int <b>Value:</b> 1
Constants.FV_XML_USEBOM_NO	int <b>Value:</b> 2
Constants.FV_XML_USEBOM_UTF8	int <b>Value:</b> 3
Constants.FV_XML_USEBOM_UTF16BE	int <b>Value:</b> 4
Constants.FV_XML_USEBOM_UTF16LE	int <b>Value:</b> 5
Constants.FV_XML_USEBOM_UTF32BE	int <b>Value:</b> 6
Constants.FV_XML_USEBOM_UTF32LE	int <b>Value:</b> 7

<b>Property name</b>	<b>Data Type</b>
Constants.FP_XmlWellFormed	int <b>Value:</b> 2806
Constants.FV_XML_WELLFORMED_YES	int <b>Value:</b> 1
Constants.FV_XML_WELLFORMED_NO	int <b>Value:</b> 2
Constants.FP_XmlFileEncoding	int <b>Value:</b> 2809
Constants.FP_XmlDocType	int <b>Value:</b> 2810
Constants.FP_XmlPublicId	int <b>Value:</b> 2811
Constants.FP_XmlSystemId	int <b>Value:</b> 2812
Constants.FP_XmlLock	int <b>Value:</b> 2841
Constants.FP_ServerUrl	int <b>Value:</b> 2807
Constants.FP_ServerState	int <b>Value:</b> 2808
Constants.FV_URL_CHECKED_IN	int <b>Value:</b> 1
Constants.FV_URL_CHECKED_OUT	int <b>Value:</b> 2
Constants.FP_KeyCatalog	int <b>Value:</b> 2856
Constants.FP_KeyCatalogType	int <b>Value:</b> 2857
Constants.FP_SpecifiedKeyCatalog	int <b>Value:</b> 2858
Constants.FP_NextOpenBookInSession	int <b>Value:</b> 490
Constants.FP_FirstComponentInBook	int <b>Value:</b> 491
Constants.FP_BooksModified	int <b>Value:</b> 492

<b>Property name</b>	<b>Data Type</b>
Constants.FP_FirstSelectedComponentInBook	int <b>Value:</b> 493
Constants.FP_BookDontUpdateReferences	int <b>Value:</b> 494
Constants.FP_BooksViewOnly	int <b>Value:</b> 497
Constants.FP_BooksSelected	int <b>Value:</b> 499
Constants.FP_TypeOfDisplayText	int <b>Value:</b> 500
Constants.FV_BK_FILENAME	int <b>Value:</b> 1
Constants.FV_BK_TEXT	int <b>Value:</b> 2
Constants.FP_BookParent	int <b>Value:</b> 515
Constants.FP_ExtractTags	int <b>Value:</b> 516
Constants.FP_GenerateInclude	int <b>Value:</b> 517
Constants.FP_ImportFmtInclude	int <b>Value:</b> 518
Constants.FP_PrintInclude	int <b>Value:</b> 519
Constants.FP_BookComponentIsGeneratable	int <b>Value:</b> 520
Constants.FP_PagePrefix	int <b>Value:</b> 521
Constants.FP_PageSuffix	int <b>Value:</b> 522
Constants.FP_PageSide	int <b>Value:</b> 523
Constants.FV_BK_START_FROM_FILE	int <b>Value:</b> 0x01
Constants.FV_BK_START_NEXT_AVAILABLE	int <b>Value:</b> 0x02

<b>Property name</b>	<b>Data Type</b>
Constants.FV_BK_START_LEFT	int <b>Value:</b> 0x03
Constants.FV_BK_START_RIGHT	int <b>Value:</b> 0x04
Constants.FP_PageNumComputeMethod	int <b>Value:</b> 524
Constants.FP_PgfNumComputeMethod	int <b>Value:</b> 525
Constants.FP_PrevComponentInBook	int <b>Value:</b> 526
Constants.FP_NextComponentInBook	int <b>Value:</b> 527
Constants.FP_InsertLinks	int <b>Value:</b> 528
Constants.FP_ComponentsIsSelected	int <b>Value:</b> 529
Constants.FP_NextSelectedComponentInBook	int <b>Value:</b> 530
Constants.FP_BookComponentType	int <b>Value:</b> 531
Constants.FV_BK_TOC	int <b>Value:</b> 0
Constants.FV_BK_LIST_FIGURE	int <b>Value:</b> 1
Constants.FV_BK_LIST_TABLE	int <b>Value:</b> 2
Constants.FV_BK_LIST_PGF	int <b>Value:</b> 3
Constants.FV_BK_LIST_MARKER	int <b>Value:</b> 4
Constants.FV_BK_LIST_MARKER_ALPHA	int <b>Value:</b> 5
Constants.FV_BK_LIST_PGF_ALPHA	int <b>Value:</b> 6
Constants.FV_BK_INDEX_STAN	int <b>Value:</b> 7

<b>Property name</b>	<b>Data Type</b>
Constants.FV_BK_INDEX_AUTHOR	int <b>Value: 8</b>
Constants.FV_BK_INDEX_SUBJECT	int <b>Value: 9</b>
Constants.FV_BK_INDEX_MARKER	int <b>Value: 10</b>
Constants.FV_BK_LIST_FORMATS	int <b>Value: 11</b>
Constants.FV_BK_LIST_REFERENCES	int <b>Value: 12</b>
Constants.FV_BK_INDEX_FORMATS	int <b>Value: 13</b>
Constants.FV_BK_INDEX_REFERENCES	int <b>Value: 14</b>
Constants.FV_BK_NUM_BOOK_LIST	int <b>Value: 15</b>
Constants.FV_BK_NOT_GENERATABLE	int <b>Value: 16</b>
Constants.FP_ComponentDisplayText	int <b>Value: 532</b>
Constants.FP_FirstComponentInBookComponent	int <b>Value: 2324</b>
Constants.FP_BookComponentParent	int <b>Value: 2325</b>
Constants.FP_ExcludeBookComponent	int <b>Value: 2326</b>
Constants.FP_BookComponentTemplatePath	int <b>Value: 2327</b>
Constants.FP_BookComponentTitle	int <b>Value: 2328</b>
Constants.FP_ComponentType	int <b>Value: 2329</b>
Constants.FP_ComponentsDitaMap	int <b>Value: 2331</b>
Constants.FP_CorrespondingXmlPath	int <b>Value: 1158</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_BK_GENERAL	int <b>Value:</b> 0x1
Constants.FV_BK_FOLDER	int <b>Value:</b> 0x2
Constants.FV_BK_BOOK	int <b>Value:</b> 0x4
Constants.FV_BK_FM	int <b>Value:</b> 0x8
Constants.FV_BK_MIF	int <b>Value:</b> 0x10
Constants.FV_BK_XML	int <b>Value:</b> 0x20
Constants.FV_BK_GROUP	int <b>Value:</b> 0x40
Constants.FV_BK_DITAMAP	int <b>Value:</b> 0x80
Constants.FV_BK_BOOKMAP	int <b>Value:</b> 0x100
Constants.FV_BK_FILE	int <b>Value:</b> 0x200
Constants.FP_NextBookComponentInDFSOrder	int <b>Value:</b> 2333
Constants.FP_BookComponentFileType	int <b>Value:</b> 2339
Constants.FP_PrevBookComponentInDFSOrder	int <b>Value:</b> 2340
Constants.FP_BookComponentsFolderWithTemplate	int <b>Value:</b> 2341
Constants.FP_BookComponentsFolderWithoutTemplate	int <b>Value:</b> 2342
Constants.FP_XmlApplicationForBookComponent	int <b>Value:</b> 2330
Constants.FP_BookComponentSequenceLevel	int <b>Value:</b> 2353
Constants.FP_DocSequenceLevel	int <b>Value:</b> 2356



<b>Property name</b>	<b>Data Type</b>
Constants.FA_COMPONENT_MOVEUP	int <b>Value: 1</b>
Constants.FA_COMPONENT_MOVEDOWN	int <b>Value: 2</b>
Constants.FA_COMPONENT_PROMOTE	int <b>Value: 3</b>
Constants.FA_COMPONENT_DEMOTE	int <b>Value: 4</b>
Constants.FP_PgfAlignment	int <b>Value: 547</b>
Constants.FV_PGF_LEFT	int <b>Value: 0x01</b>
Constants.FV_PGF_RIGHT	int <b>Value: 0x02</b>
Constants.FV_PGF_CENTER	int <b>Value: 0x03</b>
Constants.FV_PGF_JUSTIFIED	int <b>Value: 0x04</b>
Constants.FP_AutoNumString	int <b>Value: 548</b>
Constants.FP_AutoNumChar	int <b>Value: 549</b>
Constants.FP_LetterSpace	int <b>Value: 550</b>
Constants.FP_KeepWithPrev	int <b>Value: 551</b>
Constants.FP_NextTag	int <b>Value: 552</b>
Constants.FP_SpaceAbove	int <b>Value: 553</b>
Constants.FP_TopSeparator	int <b>Value: 554</b>
Constants.FP_LeftIndent	int <b>Value: 555</b>
Constants.FP_FirstIndent	int <b>Value: 556</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_OptSpace	int <b>Value:</b> 557
Constants.FP_Leading	int <b>Value:</b> 558
Constants.FP_AdjHyphens	int <b>Value:</b> 559
Constants.FP_HyphMinSuffix	int <b>Value:</b> 560
Constants.FP_BlockLines	int <b>Value:</b> 561
Constants.FP_PgflsAutoNum	int <b>Value:</b> 562
Constants.FP_NumAtEnd	int <b>Value:</b> 563
Constants.FP_Hyphenate	int <b>Value:</b> 564
Constants.FP_KeepWithNext	int <b>Value:</b> 565
Constants.FP_UseNextTag	int <b>Value:</b> 566
Constants.FP_Start	int <b>Value:</b> 567
Constants.FV_PGF_ANYWHERE	int <b>Value:</b> 0x00
Constants.FV_PGF_TOP_OF_COL	int <b>Value:</b> 0x01
Constants.FV_PGF_TOP_OF_PAGE	int <b>Value:</b> 0x02
Constants.FV_PGF_TOP_OF_LEFT_PAGE	int <b>Value:</b> 0x03
Constants.FV_PGF_TOP_OF_RIGHT_PAGE	int <b>Value:</b> 0x04
Constants.FP_SpaceBelow	int <b>Value:</b> 568
Constants.FP_BottomSeparator	int <b>Value:</b> 569

<b>Property name</b>	<b>Data Type</b>
Constants.FP_RightIndent	int <b>Value:</b> 570
Constants.FP_MinSpace	int <b>Value:</b> 571
Constants.FP_MaxSpace	int <b>Value:</b> 572
Constants.FP_NumTabs	int <b>Value:</b> 573
Constants.FP_Tabs	int <b>Value:</b> 574
Constants.FP_HyphMinPrefix	int <b>Value:</b> 575
Constants.FP_HyphMinWord	int <b>Value:</b> 576
Constants.FP_Language	int <b>Value:</b> 577
Constants.FV_LANG_NOLANGUAGE	int <b>Value:</b> 0x00
Constants.FV_LANG_ENGLISH	int <b>Value:</b> 0x01
Constants.FV_LANG_BRITISH	int <b>Value:</b> 0x02
Constants.FV_LANG_GERMAN	int <b>Value:</b> 0x03
Constants.FV_LANG_SWISS_GERMAN	int <b>Value:</b> 0x04
Constants.FV_LANG_FRENCH	int <b>Value:</b> 0x05
Constants.FV_LANG_CANADIAN_FRENCH	int <b>Value:</b> 0x06
Constants.FV_LANG_SPANISH	int <b>Value:</b> 0x07
Constants.FV_LANG_CATALAN	int <b>Value:</b> 0x08
Constants.FV_LANG_ITALIAN	int <b>Value:</b> 0x09

<b>Property name</b>	<b>Data Type</b>
Constants.FV_LANG_PORTUGUESE	int <b>Value:</b> 0x0A
Constants.FV_LANG_BRAZILIAN	int <b>Value:</b> 0x0B
Constants.FV_LANG_DANISH	int <b>Value:</b> 0x0C
Constants.FV_LANG_DUTCH	int <b>Value:</b> 0x0D
Constants.FV_LANG_NORWEGIAN	int <b>Value:</b> 0x0E
Constants.FV_LANG_NYNORSK	int <b>Value:</b> 0x0F
Constants.FV_LANG_FINNISH	int <b>Value:</b> 0x10
Constants.FV_LANG_SWEDISH	int <b>Value:</b> 0x11
Constants.FV_LANG_JAPANESE	int <b>Value:</b> 0x12
Constants.FV_LANG_TRADITIONAL_CHINESE	int <b>Value:</b> 0x13
Constants.FV_LANG_SIMPLIFIED_CHINESE	int <b>Value:</b> 0x14
Constants.FV_LANG_KOREAN	int <b>Value:</b> 0x15
Constants.FV_LANG_NEW_GERMAN	int <b>Value:</b> 0x16
Constants.FV_LANG_NEW_SWISS_GERMAN	int <b>Value:</b> 0x17
Constants.FV_LANG_NEW_DUTCH	int <b>Value:</b> 0x18
Constants.FV_LANG_GREEK	int <b>Value:</b> 0x19
Constants.FV_LANG_RUSSIAN	int <b>Value:</b> 0x1A
Constants.FV_LANG_CZECH	int <b>Value:</b> 0x1B

<b>Property name</b>	<b>Data Type</b>
Constants.FV_LANG_POLISH	int <b>Value:</b> 0x1C
Constants.FV_LANG_HUNGARIAN	int <b>Value:</b> 0x1D
Constants.FV_LANG_TURKISH	int <b>Value:</b> 0x1E
Constants.FV_LANG_SLOVAK	int <b>Value:</b> 0x1F
Constants.FV_LANG_SLOVENIAN	int <b>Value:</b> 0x20
Constants.FV_LANG_BULGARIAN	int <b>Value:</b> 0x21
Constants.FV_LANG_CROATIAN	int <b>Value:</b> 0x22
Constants.FV_LANG_ESTONIAN	int <b>Value:</b> 0x23
Constants.FV_LANG_LATVIAN	int <b>Value:</b> 0x24
Constants.FV_LANG_LITHUANIAN	int <b>Value:</b> 0x25
Constants.FV_LANG_ROMANIAN	int <b>Value:</b> 0x26
Constants.FV_LANG_NUM	int <b>Value:</b> 0x27
Constants.FP_RunInSeparator	int <b>Value:</b> 578
Constants.FP_Placement	int <b>Value:</b> 579
Constants.FV_PGF_SIDEBODY	int <b>Value:</b> 0
Constants.FV_PGF_SIDEHEAD_TOP	int <b>Value:</b> 1
Constants.FV_PGF_SIDEHEAD_FIRST_BASELINE	int <b>Value:</b> 2
Constants.FV_PGF_SIDEHEAD_LAST_BASELINE	int <b>Value:</b> 3

## Object Reference

Property name	Data Type
Constants.FV_PGF_RUN_IN	int <b>Value:</b> 4
Constants.FV_PGF_STRADDLE	int <b>Value:</b> 5
Constants.FV_PGF_STRADDLE_NORMAL_ONLY	int <b>Value:</b> 6
Constants.FP_NextPgFmtInDoc	int <b>Value:</b> 580
Constants.FP_CellTopMargin	int <b>Value:</b> 581
Constants.FP_CellBottomMargin	int <b>Value:</b> 582
Constants.FP_CellLeftMargin	int <b>Value:</b> 583
Constants.FP_CellRightMargin	int <b>Value:</b> 584
Constants.FP_CellVAlignment	int <b>Value:</b> 585
Constants.FV_PGF_V_ALIGN_TOP	int <b>Value:</b> 0
Constants.FV_PGF_V_ALIGN_MIDDLE	int <b>Value:</b> 1
Constants.FV_PGF_V_ALIGN_BOTTOM	int <b>Value:</b> 2
Constants.FP_CellMarginsFixed	int <b>Value:</b> 586
Constants.FV_PGF_FIXED_L_MARGIN	int <b>Value:</b> 0x01
Constants.FV_PGF_FIXED_B_MARGIN	int <b>Value:</b> 0x02
Constants.FV_PGF_FIXED_R_MARGIN	int <b>Value:</b> 0x04
Constants.FV_PGF_FIXED_T_MARGIN	int <b>Value:</b> 0x08
Constants.FP_LineSpacing	int <b>Value:</b> 587

<b>Property name</b>	<b>Data Type</b>
Constants.FV_PGF_FIXED	int <b>Value:</b> 0x00
Constants.FV_PGF_PROPORTIONAL	int <b>Value:</b> 0x01
Constants.FV_PGF_FLOATING	int <b>Value:</b> 0x02
Constants.FP_Locked	int <b>Value:</b> 588
Constants.FP_AcrobatLevel	int <b>Value:</b> 589
Constants.FP_PDFStructureLevel	int <b>Value:</b> 600
Constants.FP_FormatOverride	int <b>Value:</b> 590
Constants.FP_MinJRomSpace	int <b>Value:</b> 591
Constants.FP_OptJRomSpace	int <b>Value:</b> 592
Constants.FP_MaxJRomSpace	int <b>Value:</b> 593
Constants.FP_MinJLetSpace	int <b>Value:</b> 594
Constants.FP_OptJLetSpace	int <b>Value:</b> 595
Constants.FP_MaxJLetSpace	int <b>Value:</b> 596
Constants.FP_YakumonoType	int <b>Value:</b> 597
Constants.FV_FLOATING_YAKUMONO	int <b>Value:</b> 0x00
Constants.FV_MONOSPACE_YAKUMONO	int <b>Value:</b> 0x01
Constants.FV_FIXED_YAKUMONO	int <b>Value:</b> 0x02
Constants.FP_DialogEncodingName	int <b>Value:</b> 598

<b>Property name</b>	<b>Data Type</b>
Constants.FP_FMInterfaceEncodingName	int <b>Value:</b> 599
Constants.FP_CharTag	int <b>Value:</b> 602
Constants.FP_NextCharFmtInDoc	int <b>Value:</b> 603
Constants.FP_FontFamily	int <b>Value:</b> 604
Constants.FP_FontVariation	int <b>Value:</b> 605
Constants.FP_FontWeight	int <b>Value:</b> 606
Constants.FP_FontAngle	int <b>Value:</b> 607
Constants.FP_Underlining	int <b>Value:</b> 608
Constants.FV_CB_NO_UNDERLINE	int <b>Value:</b> 0
Constants.FV_CB_SINGLE_UNDERLINE	int <b>Value:</b> 1
Constants.FV_CB_DOUBLE_UNDERLINE	int <b>Value:</b> 2
Constants.FV_CB_NUMERIC_UNDERLINE	int <b>Value:</b> 3
Constants.FP_Strikethrough	int <b>Value:</b> 609
Constants.FP_Overline	int <b>Value:</b> 610
Constants.FP_ChangeBar	int <b>Value:</b> 611
Constants.FP_Outline	int <b>Value:</b> 612
Constants.FP_Shadow	int <b>Value:</b> 613
Constants.FP_PairKern	int <b>Value:</b> 614



<b>Property name</b>	<b>Data Type</b>
Constants.FP_FontSize	int <b>Value:</b> 615
Constants.FP_KernX	int <b>Value:</b> 616
Constants.FP_KernY	int <b>Value:</b> 617
Constants.FP_Spread	int <b>Value:</b> 618
Constants.FP_Capitalization	int <b>Value:</b> 619
Constants.FV_CAPITAL_CASE_NORM	int <b>Value:</b> 0
Constants.FV_CAPITAL_CASE_SMALL	int <b>Value:</b> 1
Constants.FV_CAPITAL_CASE_LOWER	int <b>Value:</b> 2
Constants.FV_CAPITAL_CASE_UPPER	int <b>Value:</b> 3
Constants.FP_Position	int <b>Value:</b> 620
Constants.FV_POS_NORM	int <b>Value:</b> 0
Constants.FV_POS_SUPER	int <b>Value:</b> 1
Constants.FV_POS_SUB	int <b>Value:</b> 2
Constants.FP_BkColor	int <b>Value:</b> 992
Constants.FP_ReviewerName	int <b>Value:</b> 993
Constants.FP_ReviewTime	int <b>Value:</b> 994
Constants.FP_UseFontFamily	int <b>Value:</b> 621
Constants.FP_UseFontVariation	int <b>Value:</b> 622

<b>Property name</b>	<b>Data Type</b>
Constants.FP_UseFontWeight	int <b>Value:</b> 623
Constants.FP_UseFontAngle	int <b>Value:</b> 624
Constants.FP_UseUnderlining	int <b>Value:</b> 625
Constants.FP_UseStrikethrough	int <b>Value:</b> 626
Constants.FP_UseOverline	int <b>Value:</b> 627
Constants.FP_UseChangeBar	int <b>Value:</b> 628
Constants.FP_UseOutline	int <b>Value:</b> 629
Constants.FP_UseShadow	int <b>Value:</b> 630
Constants.FP_UsePairKern	int <b>Value:</b> 631
Constants.FP_UseFontSize	int <b>Value:</b> 632
Constants.FP_UseKernX	int <b>Value:</b> 633
Constants.FP_UseKernY	int <b>Value:</b> 634
Constants.FP_UseSpread	int <b>Value:</b> 635
Constants.FP_UseCapitalization	int <b>Value:</b> 636
Constants.FP_UsePosition	int <b>Value:</b> 637
Constants.FP_UseColor	int <b>Value:</b> 638
Constants.FP_UseBkColor	int <b>Value:</b> 991
Constants.FP_FontPlatformName	int <b>Value:</b> 639

<b>Property name</b>	<b>Data Type</b>
Constants.FP_FontPostScriptName	int <b>Value:</b> 640
Constants.FP_FontPanoseName	int <b>Value:</b> 641
Constants.FP_FontEncodingName	int <b>Value:</b> 642
Constants.FP_Stretch	int <b>Value:</b> 643
Constants.FP_UseStretch	int <b>Value:</b> 644
Constants.FP_UseLanguage	int <b>Value:</b> 645
Constants.FP_WesternFontPlatformName	int <b>Value:</b> 646
Constants.FP_WesternFontPostScriptName	int <b>Value:</b> 647
Constants.FP_WesternFontPanoseName	int <b>Value:</b> 648
Constants.FP_Tsume	int <b>Value:</b> 649
Constants.FP_UseTsume	int <b>Value:</b> 650
Constants.FV_TAB_LEFT	int <b>Value:</b> 0x1
Constants.FV_TAB_CENTER	int <b>Value:</b> 0x2
Constants.FV_TAB_RIGHT	int <b>Value:</b> 0x3
Constants.FV_TAB_DECIMAL	int <b>Value:</b> 0x4
Constants.FV_TAB_RELATIVE_LEFT	int <b>Value:</b> 0x5
Constants.FV_TAB_RELATIVE_CENTER	int <b>Value:</b> 0x6
Constants.FV_TAB_RELATIVE_RIGHT	int <b>Value:</b> 0x7

<b>Property name</b>	<b>Data Type</b>
Constants.FV_TAB_RELATIVE_DECIMAL	int <b>Value:</b> 0x8
Constants.FP_PageNum	int <b>Value:</b> 687
Constants.FP_PointPageNum	int <b>Value:</b> 688
Constants.FP_PageBackground	int <b>Value:</b> 689
Constants.FV_BGD_DEFAULT	int <b>Value:</b> 0x00
Constants.FV_BGD_NONE	int <b>Value:</b> 0x01
Constants.FV_BGD_OTHER	int <b>Value:</b> 0x02
Constants.FP_MasterPage	int <b>Value:</b> 690
Constants.FP_PageWidth	int <b>Value:</b> 691
Constants.FP_PageHeight	int <b>Value:</b> 692
Constants.FP_PagePrev	int <b>Value:</b> 693
Constants.FP_PageNext	int <b>Value:</b> 694
Constants.FP_PageFrame	int <b>Value:</b> 695
Constants.FP_PageNumString	int <b>Value:</b> 696
Constants.FP_PagelsRecto	int <b>Value:</b> 697
Constants.FP_NextPgflnDoc	int <b>Value:</b> 716
Constants.FP_NextPgflnFlow	int <b>Value:</b> 717
Constants.FP_PrevPgflnFlow	int <b>Value:</b> 718

<b>Property name</b>	<b>Data Type</b>
Constants.FP_PgfNumber	int <b>Value:</b> 719
Constants.FP_PgfSpellChecked	int <b>Value:</b> 720
Constants.FP_PgfSplit	int <b>Value:</b> 721
Constants.FP_PgfMarkedForNamedDestination	int <b>Value:</b> 722
Constants.FP_BorderWidth	int <b>Value:</b> 740
Constants.FP_Fill	int <b>Value:</b> 741
Constants.FV_FILL_BLACK	int <b>Value:</b> 0
Constants.FV_FILL_WHITE	int <b>Value:</b> 7
Constants.FV_FILL_CLEAR	int <b>Value:</b> 15
Constants.FP_Pen	int <b>Value:</b> 742
Constants.FP_HeadArrow	int <b>Value:</b> 743
Constants.FP_TailArrow	int <b>Value:</b> 744
Constants.FP_ArrowTipAngle	int <b>Value:</b> 745
Constants.FP_ArrowBaseAngle	int <b>Value:</b> 746
Constants.FP_ArrowScaleFactor	int <b>Value:</b> 747
Constants.FP_ArrowLength	int <b>Value:</b> 748
Constants.FP_ArrowType	int <b>Value:</b> 749
Constants.FV_ARROW_STICK	int <b>Value:</b> 0x1

<b>Property name</b>	<b>Data Type</b>
Constants.FV_ARROW_HOLLOW	int <b>Value:</b> 0x2
Constants.FV_ARROW_FILLED	int <b>Value:</b> 0x3
Constants.FP_ArrowScaleHead	int <b>Value:</b> 750
Constants.FP_Color	int <b>Value:</b> 751
Constants.FP_Dash	int <b>Value:</b> 752
Constants.FP_LineCap	int <b>Value:</b> 753
Constants.FV_CAP_BUTT	int <b>Value:</b> 0x00
Constants.FV_CAP_ROUND	int <b>Value:</b> 0x01
Constants.FV_CAP_SQUARE	int <b>Value:</b> 0x02
Constants.FP_RunaroundGap	int <b>Value:</b> 754
Constants.FP_TintPercent	int <b>Value:</b> 755
Constants.FP_Overprint	int <b>Value:</b> 756
Constants.FV_KNOCKOUT	int <b>Value:</b> 0x00
Constants.FV_OVERPRINT	int <b>Value:</b> 0x01
Constants.FV_FROMCOLOR	int <b>Value:</b> 0x02
Constants.FP_UseBorderWidth	int <b>Value:</b> 3001
Constants.FP_UseFill	int <b>Value:</b> 3002
Constants.FP_UsePen	int <b>Value:</b> 3003

<b>Property name</b>	<b>Data Type</b>
Constants.FP_UseRunaroundGap	int <b>Value:</b> 3004
Constants.FP_UseTintPercent	int <b>Value:</b> 3005
Constants.FP_UseOverprint	int <b>Value:</b> 3006
Constants.FP_UseAngle	int <b>Value:</b> 3007
Constants.FP_UseLocX	int <b>Value:</b> 3008
Constants.FP_UseLocY	int <b>Value:</b> 3009
Constants.FP_UseWidth	int <b>Value:</b> 3010
Constants.FP_UseHeight	int <b>Value:</b> 3011
Constants.FP_UseRunaround	int <b>Value:</b> 3012
Constants.FP_UseAnchorType	int <b>Value:</b> 3013
Constants.FP_UseAFramelsFloating	int <b>Value:</b> 3014
Constants.FP_UseAFramelsCropped	int <b>Value:</b> 3015
Constants.FP_UseSideOffset	int <b>Value:</b> 3016
Constants.FP_UseBaselineOffset	int <b>Value:</b> 3017
Constants.FP_UseAlignment	int <b>Value:</b> 3018
Constants.FP_UseNumColumns	int <b>Value:</b> 3019
Constants.FP_UseColGapWidth	int <b>Value:</b> 3020
Constants.FP_UseSideHeadWidth	int <b>Value:</b> 3021

<b>Property name</b>	<b>Data Type</b>
Constants.FP_UseSideHeadGap	int <b>Value:</b> 3022
Constants.FP_UseTheta	int <b>Value:</b> 3023
Constants.FP_UseDTheta	int <b>Value:</b> 3024
Constants.FP_UseTextLineType	int <b>Value:</b> 3025
Constants.FP_UseMathSize	int <b>Value:</b> 3026
Constants.FP_UseInsetDpi	int <b>Value:</b> 3027
Constants.FP_UseColumnsAreBalanced	int <b>Value:</b> 3028
Constants.FP_UseSideHeadPlacement	int <b>Value:</b> 3029
Constants.FP_UseRadius	int <b>Value:</b> 3030
Constants.FP_UseFlowsAutoConnect	int <b>Value:</b> 3031
Constants.FP_UseFlowsPostScript	int <b>Value:</b> 3032
Constants.FP_FirstGraphicsFmtInDoc	int <b>Value:</b> 3101
Constants.FP_NextGraphicsFmtInDoc	int <b>Value:</b> 3102
Constants.FP_StyleTag	int <b>Value:</b> 770
Constants.FP_GraphicsSelected	int <b>Value:</b> 771
Constants.FP_GraphicCantBeSelected	int <b>Value:</b> 772
Constants.FP_GraphicsButton	int <b>Value:</b> 773
Constants.FP_FrameParent	int <b>Value:</b> 774



<b>Property name</b>	<b>Data Type</b>
Constants.FP_PrevGraphicInFrame	int <b>Value:</b> 775
Constants.FP_NextGraphicInFrame	int <b>Value:</b> 776
Constants.FP_GroupParent	int <b>Value:</b> 777
Constants.FP_PrevGraphicInGroup	int <b>Value:</b> 778
Constants.FP_NextGraphicInGroup	int <b>Value:</b> 779
Constants.FP_Angle	int <b>Value:</b> 780
Constants.FP_LocX	int <b>Value:</b> 781
Constants.FP_LocY	int <b>Value:</b> 782
Constants.FP_Width	int <b>Value:</b> 783
Constants.FP_Height	int <b>Value:</b> 784
Constants.FP_NextSelectedGraphicInDoc	int <b>Value:</b> 785
Constants.FP_NextGraphicInDoc	int <b>Value:</b> 786
Constants.FP_Runaround	int <b>Value:</b> 787
Constants.FV_TR_NONE	int <b>Value:</b> 0x01
Constants.FV_TR_CONTOUR	int <b>Value:</b> 0x02
Constants.FV_TR_BBOX	int <b>Value:</b> 0x03
Constants.FP_DesktopX	int <b>Value:</b> 788
Constants.FP_DesktopY	int <b>Value:</b> 789

<b>Property name</b>	<b>Data Type</b>
Constants.FP_DesktopWidth	int <b>Value:</b> 790
Constants.FP_DesktopHeight	int <b>Value:</b> 791
Constants.FP_IsHotspot	int <b>Value:</b> 2875
Constants.FP_HotspotCmdStr	int <b>Value:</b> 2876
Constants.FP_HotspotTitle	int <b>Value:</b> 2877
Constants.FP_FirstGraphicInGroup	int <b>Value:</b> 807
Constants.FP_LastGraphicInGroup	int <b>Value:</b> 808
Constants.FP_DTheta	int <b>Value:</b> 830
Constants.FP_Theta	int <b>Value:</b> 831
Constants.FP_InsetEditor	int <b>Value:</b> 850
Constants.FP_InsetUpdater	int <b>Value:</b> 851
Constants.FP_InsetFile	int <b>Value:</b> 852
Constants.FP_InsetFileOrigName	int <b>Value:</b> 2814
Constants.FP_InsetDpi	int <b>Value:</b> 853
Constants.FP_InsetIsFixedSize	int <b>Value:</b> 854
Constants.FP_InsetIsFlippedSideways	int <b>Value:</b> 855
Constants.FP_InsetIsInverted	int <b>Value:</b> 856
Constants.FP_InsetRasterDpi	int <b>Value:</b> 857

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

<b>Property name</b>	<b>Data Type</b>
Constants.FV_ANCHOR_TOP	int <b>Value: 2</b>
Constants.FV_ANCHOR_BELOW	int <b>Value: 3</b>
Constants.FV_ANCHOR_BOTTOM	int <b>Value: 4</b>
Constants.FV_ANCHOR_SUBCOL_LEFT	int <b>Value: 5</b>
Constants.FV_ANCHOR_SUBCOL_RIGHT	int <b>Value: 6</b>
Constants.FV_ANCHOR_SUBCOL_NEAREST	int <b>Value: 7</b>
Constants.FV_ANCHOR_SUBCOL_FARTHEST	int <b>Value: 8</b>
Constants.FV_ANCHOR_SUBCOL_INSIDE	int <b>Value: 9</b>
Constants.FV_ANCHOR_SUBCOL_OUTSIDE	int <b>Value: 10</b>
Constants.FV_ANCHOR_TEXTFRAME_LEFT	int <b>Value: 11</b>
Constants.FV_ANCHOR_TEXTFRAME_RIGHT	int <b>Value: 12</b>
Constants.FV_ANCHOR_TEXTFRAME_NEAREST	int <b>Value: 13</b>
Constants.FV_ANCHOR_TEXTFRAME_FARTHEST	int <b>Value: 14</b>
Constants.FV_ANCHOR_TEXTFRAME_INSIDE	int <b>Value: 15</b>
Constants.FV_ANCHOR_TEXTFRAME_OUTSIDE	int <b>Value: 16</b>
Constants.FV_ANCHOR_RUN_INTO_PARAGRAPH	int <b>Value: 17</b>
Constants.FP_AFramelsFloating	int <b>Value: 876</b>
Constants.FP_SideOffset	int <b>Value: 877</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_AFramelsCropped	int <b>Value: 878</b>
Constants.FP_TextLoc	int <b>Value: 879</b>
Constants.FP_PageFramePage	int <b>Value: 880</b>
Constants.FP_BaselineOffset	int <b>Value: 881</b>
Constants.FP_FirstGraphicInFrame	int <b>Value: 882</b>
Constants.FP_LastGraphicInFrame	int <b>Value: 883</b>
Constants.FP_PrevAFrame	int <b>Value: 884</b>
Constants.FP_NextAFrame	int <b>Value: 885</b>
Constants.FP_Alignment	int <b>Value: 886</b>
Constants.FV_ALIGN_LEFT	int <b>Value: 0</b>
Constants.FV_ALIGN_CENTER	int <b>Value: 1</b>
Constants.FV_ALIGN_RIGHT	int <b>Value: 2</b>
Constants.FV_ALIGN_INSIDE	int <b>Value: 3</b>
Constants.FV_ALIGN_OUTSIDE	int <b>Value: 4</b>
Constants.FP_MifFileEntity	int <b>Value: 2828</b>
Constants.FP_MifFileName	int <b>Value: 2829</b>
Constants.FP_MifFileOrigFormat	int <b>Value: 2878</b>
Constants.FP_PolyIsBezier	int <b>Value: 906</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_NumPoints	int <b>Value: 907</b>
Constants.FP_Points	int <b>Value: 908</b>
Constants.FP_RectangleIsSmoothed	int <b>Value: 929</b>
Constants.FP_Radius	int <b>Value: 950</b>
Constants.FP_FirstPgf	int <b>Value: 972</b>
Constants.FP_LastPgf	int <b>Value: 973</b>
Constants.FP_FirstAFrame	int <b>Value: 974</b>
Constants.FP_LastAFrame	int <b>Value: 975</b>
Constants.FP_FirstFn	int <b>Value: 976</b>
Constants.FP_LastFn	int <b>Value: 977</b>
Constants.FP_FirstCell	int <b>Value: 978</b>
Constants.FP_LastCell	int <b>Value: 979</b>
Constants.FP_PrevTextFrameInFlow	int <b>Value: 980</b>
Constants.FP_NextTextFrameInFlow	int <b>Value: 981</b>
Constants.FP_Flow	int <b>Value: 982</b>
Constants.FP_NumColumns	int <b>Value: 984</b>
Constants.FP_ColGapWidth	int <b>Value: 985</b>
Constants.FP_FirstSubCol	int <b>Value: 986</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_LastSubCol	int <b>Value:</b> 987
Constants.FP_LineSpacingFactor	int <b>Value:</b> 988
Constants.FP_PgfSpacingFactor	int <b>Value:</b> 989
Constants.FP_CellMarginSpacingDelta	int <b>Value:</b> 990
Constants.FP_SideHeadPlacement	int <b>Value:</b> 1171
Constants.FV_SH_LEFT	int <b>Value:</b> 0x00
Constants.FV_SH_RIGHT	int <b>Value:</b> 0x01
Constants.FV_SH_INSIDE	int <b>Value:</b> 0x02
Constants.FV_SH_OUTSIDE	int <b>Value:</b> 0x03
Constants.FP_SideHeadWidth	int <b>Value:</b> 1172
Constants.FP_SideHeadGap	int <b>Value:</b> 1173
Constants.FP_ColumnsAreBalanced	int <b>Value:</b> 1175
Constants.FP_InTextObj	int <b>Value:</b> 1000
Constants.FP_InTextFrame	int <b>Value:</b> 1001
Constants.FP_FnNum	int <b>Value:</b> 1002
Constants.FP_NextFnInDoc	int <b>Value:</b> 1003
Constants.FP_PrevFn	int <b>Value:</b> 1004
Constants.FP_NextFn	int <b>Value:</b> 1005

<b>Property name</b>	<b>Data Type</b>
Constants.FP_FnAnchorString	int <b>Value:</b> 1006
Constants.FP_OldTypeNum	int <b>Value:</b> 1024
Constants.FP_MarkerText	int <b>Value:</b> 1025
Constants.FP_NextMarkerInDoc	int <b>Value:</b> 1026
Constants.FP_MarkerTypeid	int <b>Value:</b> 1027
Constants.FP_NextMarkerTypeInDoc	int <b>Value:</b> 1028
Constants.FP_InvariantName	int <b>Value:</b> 1029
Constants.FP_Public	int <b>Value:</b> 1030
Constants.FP_Transient	int <b>Value:</b> 1031
Constants.FP_Required	int <b>Value:</b> 1032
Constants.FP_NextVarInDoc	int <b>Value:</b> 1046
Constants.FP_VarFmt	int <b>Value:</b> 1047
Constants.FP_SystemVar	int <b>Value:</b> 1049
Constants.FV_VAR_USER_VARIABLE	int <b>Value:</b> 0
Constants.FV_VAR_CURRENT_PAGE_NUM	int <b>Value:</b> 1
Constants.FV_VAR_PAGE_COUNT	int <b>Value:</b> 2
Constants.FV_VAR_CURRENT_DATE_LONG	int <b>Value:</b> 3
Constants.FV_VAR_CURRENT_DATE_SHORT	int <b>Value:</b> 4



<b>Property name</b>	<b>Data Type</b>
Constants.FV_VAR_MODIFICATION_DATE_LONG	int <b>Value: 5</b>
Constants.FV_VAR_MODIFICATION_DATE_SHORT	int <b>Value: 6</b>
Constants.FV_VAR_CREATION_DATE_LONG	int <b>Value: 7</b>
Constants.FV_VAR_CREATION_DATE_SHORT	int <b>Value: 8</b>
Constants.FV_VAR_FILE_NAME_LONG	int <b>Value: 9</b>
Constants.FV_VAR_FILE_NAME_SHORT	int <b>Value: 10</b>
Constants.FV_VAR_HEADER_FOOTER_1	int <b>Value: 11</b>
Constants.FV_VAR_HEADER_FOOTER_2	int <b>Value: 12</b>
Constants.FV_VAR_HEADER_FOOTER_3	int <b>Value: 13</b>
Constants.FV_VAR_HEADER_FOOTER_4	int <b>Value: 14</b>
Constants.FV_VAR_TABLE_CONTINUATION	int <b>Value: 15</b>
Constants.FV_VAR_TABLE_SHEET	int <b>Value: 16</b>
Constants.FV_VAR_HEADER_FOOTER_5	int <b>Value: 19</b>
Constants.FV_VAR_HEADER_FOOTER_6	int <b>Value: 20</b>
Constants.FV_VAR_HEADER_FOOTER_7	int <b>Value: 21</b>
Constants.FV_VAR_HEADER_FOOTER_8	int <b>Value: 22</b>
Constants.FV_VAR_HEADER_FOOTER_9	int <b>Value: 23</b>
Constants.FV_VAR_HEADER_FOOTER_10	int <b>Value: 24</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_VAR_HEADER_FOOTER_11	int <b>Value: 25</b>
Constants.FV_VAR_HEADER_FOOTER_12	int <b>Value: 26</b>
Constants.FV_VAR_HEADER_FOOTER_13	int <b>Value: 27</b>
Constants.FV_VAR_HEADER_FOOTER_14	int <b>Value: 28</b>
Constants.FV_VAR_HEADER_FOOTER_15	int <b>Value: 29</b>
Constants.FV_VAR_HEADER_FOOTER_16	int <b>Value: 30</b>
Constants.FV_VAR_HEADER_FOOTER_17	int <b>Value: 31</b>
Constants.FV_VAR_HEADER_FOOTER_18	int <b>Value: 32</b>
Constants.FP_Fmt	int <b>Value: 1050</b>
Constants.FP_NextVarFmtInDoc	int <b>Value: 1051</b>
Constants.FP_XRefFile	int <b>Value: 1070</b>
Constants.FP_TextRange	int <b>Value: 1071</b>
Constants.FP_NextXRefInDoc	int <b>Value: 1072</b>
Constants.FP_XRefFmt	int <b>Value: 1073</b>
Constants.FP_XRefSrcText	int <b>Value: 1075</b>
Constants.FP_XRefSrcIsElem	int <b>Value: 1076</b>
Constants.FP_XRefsUnresolved	int <b>Value: 1077</b>
Constants.FP_XRefSrcElemNonUniqueId	int <b>Value: 1152</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_XRefAltText	int <b>Value:</b> 1153
Constants.FP_XRefClientName	int <b>Value:</b> 1154
Constants.FP_XRefClientType	int <b>Value:</b> 1155
Constants.FP_XRefResolvedWithKeyref	int <b>Value:</b> 1156
Constants.FP_XRefRetainKeydefText	int <b>Value:</b> 1157
Constants.FP_NextXRefFmtInDoc	int <b>Value:</b> 1074
Constants.FP_TextLineType	int <b>Value:</b> 1095
Constants.FV_TEXTLINE_LEFT	int <b>Value:</b> 0
Constants.FV_TEXTLINE_RIGHT	int <b>Value:</b> 1
Constants.FV_TEXTLINE_CENTER	int <b>Value:</b> 2
Constants.FV_TEXTLINE_MATH	int <b>Value:</b> 3
Constants.FP_BasePointX	int <b>Value:</b> 1116
Constants.FP_BasePointY	int <b>Value:</b> 1117
Constants.FP_NextCondFmtInDoc	int <b>Value:</b> 1138
Constants.FP_CondFmtsShown	int <b>Value:</b> 1139
Constants.FP_SepOverride	int <b>Value:</b> 1140
Constants.FP_UseSepOverride	int <b>Value:</b> 1141
Constants.FP_StyleOverride	int <b>Value:</b> 1142

<b>Property name</b>	<b>Data Type</b>
Constants.FV_CN_NO_OVERRIDE	int <b>Value: 0</b>
Constants.FV_CN_OVERLINE	int <b>Value: 1</b>
Constants.FV_CN_STRIKETHROUGH	int <b>Value: 2</b>
Constants.FV_CN_SINGLE_UNDERLINE	int <b>Value: 3</b>
Constants.FV_CN_DOUBLE_UNDERLINE	int <b>Value: 4</b>
Constants.FV_CN_CHANGEBAR	int <b>Value: 5</b>
Constants.FV_CN_NUMERIC_UNDERLINE	int <b>Value: 6</b>
Constants.FV_CN_NMRIC_AND_CHNGBAR	int <b>Value: 7</b>
Constants.FP_NextAttrCondExprInDoc	int <b>Value: 2816</b>
Constants.FP_AttrCondExprStr	int <b>Value: 2817</b>
Constants.FP_AttrCondExprIsActive	int <b>Value: 2818</b>
Constants.FP_InCond	int <b>Value: 1150</b>
Constants.FP_StyleOverrides	int <b>Value: 1151</b>
Constants.FV_CS_NO_OVERRIDE	int <b>Value: 0x00</b>
Constants.FV_CS_OVERLINE	int <b>Value: 0x01</b>
Constants.FV_CS_STRIKETHROUGH	int <b>Value: 0x02</b>
Constants.FV_CS_SINGLE_UNDERLINE	int <b>Value: 0x04</b>
Constants.FV_CS_DOUBLE_UNDERLINE	int <b>Value: 0x08</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_CS_CHANGEBAR	int <b>Value:</b> 0x10
Constants.FV_CS_NUMERIC_UNDERLINE	int <b>Value:</b> 0x20
Constants.FP_NextFlowInDoc	int <b>Value:</b> 1162
Constants.FP_FlowsSynchronized	int <b>Value:</b> 1163
Constants.FP_MinHang	int <b>Value:</b> 1164
Constants.FP_FlowsAutoConnect	int <b>Value:</b> 1165
Constants.FP_FlowsFeathered	int <b>Value:</b> 1166
Constants.FP_Spacing	int <b>Value:</b> 1167
Constants.FP_FlowsPostScript	int <b>Value:</b> 1168
Constants.FP_FirstTextFrameInFlow	int <b>Value:</b> 1169
Constants.FP_LastTextFrameInFlow	int <b>Value:</b> 1170
Constants.FP_SideHeadRoomInFlow	int <b>Value:</b> 1174
Constants.FP_MaxInterlinePadding	int <b>Value:</b> 470
Constants.FP_MaxInterPgfPadding	int <b>Value:</b> 471
Constants.FP_CellUseOverrideLRuling	int <b>Value:</b> 1190
Constants.FP_CellUseOverrideRRuling	int <b>Value:</b> 1191
Constants.FP_CellUseOverrideTRuling	int <b>Value:</b> 1192
Constants.FP_CellUseOverrideBRuling	int <b>Value:</b> 1193

<b>Property name</b>	<b>Data Type</b>
Constants.FP_CellOverrideShading	int <b>Value:</b> 1194
Constants.FP_CellOverrideFill	int <b>Value:</b> 1195
Constants.FP_CellUseOverrideFill	int <b>Value:</b> 1196
Constants.FP_CellUseOverrideShading	int <b>Value:</b> 1197
Constants.FP_CellRow	int <b>Value:</b> 1198
Constants.FP_NextCellInTbl	int <b>Value:</b> 1199
Constants.FP_CellBelowInTbl	int <b>Value:</b> 1200
Constants.FP_PrevCellInRow	int <b>Value:</b> 1201
Constants.FP_NextCellInRow	int <b>Value:</b> 1202
Constants.FP_CellAboveInCol	int <b>Value:</b> 1203
Constants.FP_CellBelowInCol	int <b>Value:</b> 1204
Constants.FP_CellColNum	int <b>Value:</b> 1205
Constants.FP_CellsStraddled	int <b>Value:</b> 1206
Constants.FP_CellNumRowsStraddled	int <b>Value:</b> 1207
Constants.FP_CellNumColsStraddled	int <b>Value:</b> 1208
Constants.FP_CellAngle	int <b>Value:</b> 1209
Constants.FP_CellOverrideTopRuling	int <b>Value:</b> 1210
Constants.FP_CellOverrideBottomRuling	int <b>Value:</b> 1211

<b>Property name</b>	<b>Data Type</b>
Constants.FP_CellOverrideLeftRuling	int <b>Value:</b> 1212
Constants.FP_CellOverrideRightRuling	int <b>Value:</b> 1213
Constants.FP_CellDefaultTopRuling	int <b>Value:</b> 1214
Constants.FP_CellDefaultBottomRuling	int <b>Value:</b> 1215
Constants.FP_CellDefaultLeftRuling	int <b>Value:</b> 1216
Constants.FP_CellDefaultRightRuling	int <b>Value:</b> 1217
Constants.FP_CellsShown	int <b>Value:</b> 1218
Constants.FP_PrevCell	int <b>Value:</b> 1219
Constants.FP_NextCell	int <b>Value:</b> 1220
Constants.FP_PrevRowInTbl	int <b>Value:</b> 1244
Constants.FP_NextRowInTbl	int <b>Value:</b> 1245
Constants.FP_RowTbl	int <b>Value:</b> 1246
Constants.FP_FirstCellInRow	int <b>Value:</b> 1247
Constants.FP_RowKeepWithNext	int <b>Value:</b> 1248
Constants.FP_RowKeepWithPrev	int <b>Value:</b> 1249
Constants.FP_RowMaxHeight	int <b>Value:</b> 1250
Constants.FP_RowMinHeight	int <b>Value:</b> 1251
Constants.FP_RowStart	int <b>Value:</b> 1252

<b>Property name</b>	<b>Data Type</b>
Constants.FV_ROW_ANYWHERE	int <b>Value: 0</b>
Constants.FV_ROW_TOP_OF_COL	int <b>Value: 1</b>
Constants.FV_ROW_TOP_OF_PAGE	int <b>Value: 2</b>
Constants.FV_ROW_TOP_OF_LEFT_PAGE	int <b>Value: 3</b>
Constants.FV_ROW_TOP_OF_RIGHT_PAGE	int <b>Value: 4</b>
Constants.FP_RowType	int <b>Value: 1253</b>
Constants.FV_ROW_HEADING	int <b>Value: 0</b>
Constants.FV_ROW_BODY	int <b>Value: 1</b>
Constants.FV_ROW_FOOTING	int <b>Value: 2</b>
Constants.FP_RowsShown	int <b>Value: 1254</b>
Constants.FP_TblTag	int <b>Value: 1275</b>
Constants.FP_NextTblFmtInDoc	int <b>Value: 1276</b>
Constants.FP_TblLeftIndent	int <b>Value: 1278</b>
Constants.FP_TblRightIndent	int <b>Value: 1279</b>
Constants.FP_TblSpaceAbove	int <b>Value: 1280</b>
Constants.FP_TblSpaceBelow	int <b>Value: 1281</b>
Constants.FP_TblAlignment	int <b>Value: 1282</b>
Constants.FV_ALIGN_TBL_LEFT	int <b>Value: 0</b>



<b>Property name</b>	<b>Data Type</b>
Constants.FV_ALIGN_TBL_CENTER	int <b>Value: 1</b>
Constants.FV_ALIGN_TBL_RIGHT	int <b>Value: 2</b>
Constants.FP_TblPlacement	int <b>Value: 1283</b>
Constants.FV_TBL_ANYWHERE	int <b>Value: 0</b>
Constants.FV_TBL_TOP_OF_COL	int <b>Value: 1</b>
Constants.FV_TBL_TOP_OF_PAGE	int <b>Value: 2</b>
Constants.FV_TBL_TOP_OF_LEFT_PAGE	int <b>Value: 3</b>
Constants.FV_TBL_TOP_OF_RIGHT_PAGE	int <b>Value: 4</b>
Constants.FV_TBL_FLOAT	int <b>Value: 5</b>
Constants.FP_TblInitNumCols	int <b>Value: 1284</b>
Constants.FP_TblInitNumHRows	int <b>Value: 1285</b>
Constants.FP_TblInitNumBodyRows	int <b>Value: 1286</b>
Constants.FP_TblInitNumFRows	int <b>Value: 1287</b>
Constants.FP_TblNumbering	int <b>Value: 1288</b>
Constants.FV_TBL_NUM_BY_ROW	int <b>Value: 0</b>
Constants.FV_TBL_NUM_BY_COL	int <b>Value: 1</b>
Constants.FP_TblTitlePosition	int <b>Value: 1289</b>
Constants.FV_TBL_NO_TITLE	int <b>Value: 0</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_TBL_TITLE_ABOVE	int <b>Value: 1</b>
Constants.FV_TBL_TITLE_BELOW	int <b>Value: 2</b>
Constants.FP_TblTitleGap	int <b>Value: 1290</b>
Constants.FP_OrphanRows	int <b>Value: 1291</b>
Constants.FP_TblCatalogEntry	int <b>Value: 1292</b>
Constants.FP_TblColRulingPeriod	int <b>Value: 1293</b>
Constants.FP_TblBodyRowRulingPeriod	int <b>Value: 1294</b>
Constants.FP_TblLastBodyRuling	int <b>Value: 1295</b>
Constants.FP_TblHFFill	int <b>Value: 1296</b>
Constants.FP_TblHFCOLOR	int <b>Value: 1297</b>
Constants.FP_TblBodyFirstFill	int <b>Value: 1298</b>
Constants.FP_TblBodyFirstColor	int <b>Value: 1299</b>
Constants.FP_TblBodyShadeBy	int <b>Value: 1300</b>
Constants.FP_TblBodyFirstPeriod	int <b>Value: 1301</b>
Constants.FP_TblBodyNextFill	int <b>Value: 1302</b>
Constants.FP_TblBodyNextPeriod	int <b>Value: 1303</b>
Constants.FP_TblBodyNextColor	int <b>Value: 1304</b>
Constants.FP_TblTopRuling	int <b>Value: 1305</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_TblBottomRuling	int <b>Value:</b> 1306
Constants.FP_TblLeftRuling	int <b>Value:</b> 1307
Constants.FP_TblRightRuling	int <b>Value:</b> 1308
Constants.FP_TblColRuling	int <b>Value:</b> 1309
Constants.FP_TblBodyRowRuling	int <b>Value:</b> 1310
Constants.FP_TblHFSeparatorRuling	int <b>Value:</b> 1311
Constants.FP_TblHFRowRuling	int <b>Value:</b> 1312
Constants.FP_TblOtherBodyRowRuling	int <b>Value:</b> 1313
Constants.FP_TblOtherColRuling	int <b>Value:</b> 1314
Constants.FP_TblCellTopMargin	int <b>Value:</b> 1315
Constants.FP_TblCellBottomMargin	int <b>Value:</b> 1316
Constants.FP_TblCellLeftMargin	int <b>Value:</b> 1317
Constants.FP_TblCellRightMargin	int <b>Value:</b> 1318
Constants.FP_TblInLockedTi	int <b>Value:</b> 1319
Constants.FP_TblNumCols	int <b>Value:</b> 1335
Constants.FP_TblNumRows	int <b>Value:</b> 1336
Constants.FP_NextTblInDoc	int <b>Value:</b> 1337
Constants.FP_FirstRowInTbl	int <b>Value:</b> 1338

<b>Property name</b>	<b>Data Type</b>
Constants.FP_LastRowInTbl	int <b>Value:</b> 1339
Constants.FP_TblWidth	int <b>Value:</b> 1340
Constants.FP_TopRowSelection	int <b>Value:</b> 1341
Constants.FP_BottomRowSelection	int <b>Value:</b> 1342
Constants.FP_LeftColNum	int <b>Value:</b> 1343
Constants.FP_RightColNum	int <b>Value:</b> 1344
Constants.FP_TblColWidths	int <b>Value:</b> 1345
Constants.FP_TblTitleSelected	int <b>Value:</b> 1346
Constants.FP_NextRulingFmtInDoc	int <b>Value:</b> 1430
Constants.FP_RulingPenWidth	int <b>Value:</b> 1432
Constants.FP_RulingGap	int <b>Value:</b> 1433
Constants.FP_RulingSep	int <b>Value:</b> 1434
Constants.FP_RulingLines	int <b>Value:</b> 1435
Constants.FP_NextColorInDoc	int <b>Value:</b> 1455
Constants.FP_Pantone	int <b>Value:</b> 1456
Constants.FP_Cyan	int <b>Value:</b> 1457
Constants.FP_Magenta	int <b>Value:</b> 1458
Constants.FP_Yellow	int <b>Value:</b> 1459

<b>Property name</b>	<b>Data Type</b>
Constants.FP_Black	int <b>Value:</b> 1460
Constants.FP_ColorViewCtl	int <b>Value:</b> 1461
Constants.FV_SEP_NORMAL	int <b>Value:</b> 0x0
Constants.FV_SEP_NONE	int <b>Value:</b> 0x1
Constants.FV_SEP_WHITE	int <b>Value:</b> 0x2
Constants.FP_ColorPrintCtl	int <b>Value:</b> 1462
Constants.FV_PRINT_SPOT	int <b>Value:</b> 0x0
Constants.FV_PRINT_PROCESS	int <b>Value:</b> 0x1
Constants.FV_PRINT_NO	int <b>Value:</b> 0x2
Constants.FP_ReservedColor	int <b>Value:</b> 1463
Constants.FV_COLOR_NOT_RESERVED	int <b>Value:</b> 0
Constants.FV_COLOR_CYAN	int <b>Value:</b> 1
Constants.FV_COLOR_MAGENTA	int <b>Value:</b> 2
Constants.FV_COLOR_YELLOW	int <b>Value:</b> 3
Constants.FV_COLOR_BLACK	int <b>Value:</b> 4
Constants.FV_COLOR_WHITE	int <b>Value:</b> 5
Constants.FV_COLOR_RED	int <b>Value:</b> 6
Constants.FV_COLOR_GREEN	int <b>Value:</b> 7

<b>Property name</b>	<b>Data Type</b>
Constants.FV_COLOR_BLUE	int <b>Value: 8</b>
Constants.FV_COLOR_DARKGREY	int <b>Value: 9</b>
Constants.FV_COLOR_PALEGREEN	int <b>Value: 10</b>
Constants.FV_COLOR_FORESTGREEN	int <b>Value: 11</b>
Constants.FV_COLOR_ROYALBLUE	int <b>Value: 12</b>
Constants.FV_COLOR_MAUVE	int <b>Value: 13</b>
Constants.FV_COLOR_LIGHTSALMON	int <b>Value: 14</b>
Constants.FV_COLOR_DARKYELLOW	int <b>Value: 15</b>
Constants.FV_COLOR_SALMON	int <b>Value: 16</b>
Constants.FP_FamilyName	int <b>Value: 1464</b>
Constants.FP_InkName	int <b>Value: 1465</b>
Constants.FP_ColorTintPercent	int <b>Value: 1466</b>
Constants.FP_TintBaseColor	int <b>Value: 1467</b>
Constants.FV_NO_BASE_COLOR	int <b>Value: 0x00</b>
Constants.FP_ColorOverprint	int <b>Value: 1468</b>
Constants.FV_COLOR_KNOCKOUT	int <b>Value: 0x00</b>
Constants.FV_COLOR_OVERPRINT	int <b>Value: 0x01</b>
Constants.FP_NextRubInDoc	int <b>Value: 1469</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_OyamojiTextRange	int <b>Value:</b> 1470
Constants.FP_RubiTextRange	int <b>Value:</b> 1471
Constants.FP_HypertextDoValidate	int <b>Value:</b> 2300
Constants.FP_HypertextCommandText	int <b>Value:</b> 2301
Constants.FP_HypertextParsedArgs	int <b>Value:</b> 2302
Constants.FP_HypertextParseErr	int <b>Value:</b> 2303
Constants.FV_HypertextSyntaxOK	int <b>Value:</b> 0
Constants.FV_HypertextEmptyCommand	int <b>Value:</b> 1
Constants.FV_HypertextUnrecognizedCommand	int <b>Value:</b> 2
Constants.FV_HypertextMissingArguments	int <b>Value:</b> 3
Constants.FV_HypertextExtraArguments	int <b>Value:</b> 4
Constants.FV_HypertextBadSyntaxPathSpec	int <b>Value:</b> 10
Constants.FV_HypertextUnanchoredPartialPath	int <b>Value:</b> 11
Constants.FV_HypertextHelpDirNotFound	int <b>Value:</b> 20
Constants.FV_HypertextExpectedANumberParam	int <b>Value:</b> 30
Constants.FP_HypertextValidateErr	int <b>Value:</b> 2304
Constants.FV_HypertextValid	int <b>Value:</b> 0
Constants.FV_HypertextUsesDefaultText	int <b>Value:</b> 200

<b>Property name</b>	<b>Data Type</b>
Constants.FV_HypertextFileNotRegular	int <b>Value:</b> 210
Constants.FV_HypertextFileNotMakerDoc	int <b>Value:</b> 211
Constants.FV_HypertextCantOpenDestFile	int <b>Value:</b> 212
Constants.FV_HypertextDestinationLinkNotFound	int <b>Value:</b> 220
Constants.FV_HypertextDuplicateLinkName	int <b>Value:</b> 221
Constants.FV_HypertextPageNameNotFound	int <b>Value:</b> 230
Constants.FV_HypertextUnrecognizedObjectType	int <b>Value:</b> 240
Constants.FV_HypertextObjectIDNotFound	int <b>Value:</b> 241
Constants.FV_HypertextBadMatrixSize	int <b>Value:</b> 250
Constants.FV_HypertextMatrixCommandInvalid	int <b>Value:</b> 251
Constants.FV_HypertextFlowMissingLines	int <b>Value:</b> 252
Constants.FV_HypertextNoNamedFlow	int <b>Value:</b> 260
Constants.FV_HypertextRecursiveFlow	int <b>Value:</b> 261
Constants.FV_HypertextMissingPopupMarker	int <b>Value:</b> 270
Constants.FV_HypertextMissingPopupLabelItem	int <b>Value:</b> 271
Constants.FV_HypertextEmptyLineInMiddleOfPopup	int <b>Value:</b> 272
Constants.FV_HypertextCommandIllegalWithinPopup	int <b>Value:</b> 273
Constants.FV_HypertextFcodeInvalid	int <b>Value:</b> 280



<b>Property name</b>	<b>Data Type</b>
Constants.FP_HypertextParseBadParam	int <b>Value:</b> 2305
Constants.FP_HypertextParseErrMsg	int <b>Value:</b> 2306
Constants.FP_HypertextParsedCmdCode	int <b>Value:</b> 2307
Constants.FV_CmdNotTyped	int <b>Value:</b> 0
Constants.FV_CmdError	int <b>Value:</b> 1
Constants.FV_CmdUnknown	int <b>Value:</b> 2
Constants.FV_CmdNoop	int <b>Value:</b> 3
Constants.FV_CmdAlert	int <b>Value:</b> 8
Constants.FV_CmdAlertTitle	int <b>Value:</b> 9
Constants.FV_CmdExit	int <b>Value:</b> 10
Constants.FV_CmdGoToLink	int <b>Value:</b> 11
Constants.FV_CmdGoToLinkFitWin	int <b>Value:</b> 12
Constants.FV_CmdGoToNew	int <b>Value:</b> 13
Constants.FV_CmdGoToPage	int <b>Value:</b> 14
Constants.FV_CmdGoToObjectId	int <b>Value:</b> 15
Constants.FV_CmdGoToObjectIdFitWin	int <b>Value:</b> 16
Constants.FV_CmdMatrix	int <b>Value:</b> 17
Constants.FV_CmdMessage	int <b>Value:</b> 18

<b>Property name</b>	<b>Data Type</b>
Constants.FV_CmdNewLink	int <b>Value: 19</b>
Constants.FV_CmdNextPage	int <b>Value: 20</b>
Constants.FV_CmdPreviousPage	int <b>Value: 21</b>
Constants.FV_CmdOpenLink	int <b>Value: 22</b>
Constants.FV_CmdOpenLinkFitWin	int <b>Value: 23</b>
Constants.FV_CmdOpenNew	int <b>Value: 24</b>
Constants.FV_CmdOpenObjectId	int <b>Value: 25</b>
Constants.FV_CmdOpenObjectIdFitWin	int <b>Value: 26</b>
Constants.FV_CmdOpenPage	int <b>Value: 27</b>
Constants.FV_CmdPopup	int <b>Value: 28</b>
Constants.FV_CmdPreviousLink	int <b>Value: 29</b>
Constants.FV_CmdPreviousLinkFitWin	int <b>Value: 30</b>
Constants.FV_CmdQuit	int <b>Value: 31</b>
Constants.FV_CmdQuitAll	int <b>Value: 32</b>
Constants.FV_CmdMultiMedia	int <b>Value: 33</b>
Constants.FV_CmdFCodes	int <b>Value: 256</b>
Constants.FV_CmdOutCodes	int <b>Value: 257</b>
Constants.FV_CmdInCodes	int <b>Value: 258</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_CmdApplyMathRules	int <b>Value:</b> 259
Constants.FV_CmdThesaurusLookup	int <b>Value:</b> 260
Constants.FV_CmdNative	int <b>Value:</b> 261
Constants.FV_CmdHelpLink	int <b>Value:</b> 262
Constants.FV_CmdBeginRange	int <b>Value:</b> 263
Constants.FV_CmdEndRange	int <b>Value:</b> 264
Constants.FP_HypertextParsedCmdDest	int <b>Value:</b> 2308
Constants.FV_DestNowhere	int <b>Value:</b> 0
Constants.FV_DestMarkerNewLink	int <b>Value:</b> 1
Constants.FV_DestFirstPage	int <b>Value:</b> 2
Constants.FV_DestLastPage	int <b>Value:</b> 3
Constants.FV_DestPageNum	int <b>Value:</b> 4
Constants.FV_DestFluidFlow	int <b>Value:</b> 5
Constants.FV_DestMarker	int <b>Value:</b> 6
Constants.FV_DestObjectId	int <b>Value:</b> 7
Constants.FV_DestXRef	int <b>Value:</b> 8
Constants.FP_HypertextParsedCmdDestObjType	int <b>Value:</b> 2309
Constants.FV_ObjectUnknown	int <b>Value:</b> 0

<b>Property name</b>	<b>Data Type</b>
Constants.FV_ObjectMarker	int <b>Value: 1</b>
Constants.FV_ObjectPgf	int <b>Value: 2</b>
Constants.FV_ObjectXref	int <b>Value: 3</b>
Constants.FV_ObjectGraphic	int <b>Value: 4</b>
Constants.FV_ObjectElement	int <b>Value: 5</b>
Constants.FV_ObjectTextInset	int <b>Value: 6</b>
Constants.FV_ObjectDataLink	int <b>Value: 7</b>
Constants.FP_HypertextParsedCmdDestObjID	int <b>Value: 2310</b>
Constants.FP_HypertextParsedCmdMatrixRows	int <b>Value: 2311</b>
Constants.FP_HypertextParsedCmdMatrixColumns	int <b>Value: 2312</b>
Constants.FP_HypertextParsedLinkName	int <b>Value: 2313</b>
Constants.FP_HypertextParsedPageName	int <b>Value: 2314</b>
Constants.FP_HypertextParsedFlowName	int <b>Value: 2315</b>
Constants.FP_HypertextParsedRangeName	int <b>Value: 2316</b>
Constants.FP_HypertextParsedClientName	int <b>Value: 2317</b>
Constants.FP_HypertextParsedTitle	int <b>Value: 2318</b>
Constants.FP_HypertextParsedMessage	int <b>Value: 2319</b>
Constants.FP_HypertextParsedDIFFileName	int <b>Value: 2320</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_FirstElementDefInDoc	int <b>Value:</b> 1483
Constants.FP_ElementBoundaryDisplay	int <b>Value:</b> 1484
Constants.FP_BannerTextDisplay	int <b>Value:</b> 1600
Constants.FP_BooleanConditionExpression	int <b>Value:</b> 2321
Constants.FP_BooleanConditionState	int <b>Value:</b> 2322
Constants.FP_BooleanConditionExpressionTag	int <b>Value:</b> 995
Constants.FP_IsDitamapInResourceManager	int <b>Value:</b> 2332
Constants.FP_IsDocDitamap	int <b>Value:</b> 2879
Constants.FP_IsDocDita	int <b>Value:</b> 2880
Constants.FP_DirectOLESupportInXml	int <b>Value:</b> 2881
Constants.FP_DocIsStructuredDummy	int <b>Value:</b> 2882
Constants.FP_StructuredDummyDocText	int <b>Value:</b> 2883
Constants.FV_ELEM_DISP_NONE	int <b>Value:</b> 0
Constants.FV_ELEM_DISP_BRACKETS	int <b>Value:</b> 1
Constants.FV_ELEM_DISP_TAGS	int <b>Value:</b> 2
Constants.FP_ElementCatalogDisplay	int <b>Value:</b> 1485
Constants.FV_ELCAT_STRICT	int <b>Value:</b> 0x00
Constants.FV_ELCAT_LOOSE	int <b>Value:</b> 0x01

<b>Property name</b>	<b>Data Type</b>
Constants.FV_ELCAT_CHILDREN	int <b>Value:</b> 0x02
Constants.FV_ELCAT_ALL	int <b>Value:</b> 0x03
Constants.FV_ELCAT_CUSTOM	int <b>Value:</b> 0x04
Constants.FP_DefaultInclusions	int <b>Value:</b> 1486
Constants.FP_DefaultExclusions	int <b>Value:</b> 1487
Constants.FP_ElementCatalog	int <b>Value:</b> 1488
Constants.FP_FirstFmtChangeListInDoc	int <b>Value:</b> 1489
Constants.FP_NewElemAttrDisplay	int <b>Value:</b> 1490
Constants.FV_ATTR_DISP_NONE	int <b>Value:</b> 1
Constants.FV_ATTR_DISP_REQSPEC	int <b>Value:</b> 2
Constants.FV_ATTR_DISP_ALL	int <b>Value:</b> 3
Constants.FP_NewElemAttrEditing	int <b>Value:</b> 1491
Constants.FV_ATTR_EDIT_NONE	int <b>Value:</b> 0
Constants.FV_ATTR_EDIT_REQUIRED	int <b>Value:</b> 1
Constants.FV_ATTR_EDIT_ALWAYS	int <b>Value:</b> 2
Constants.FP_UselInitialStructure	int <b>Value:</b> 1492
Constants.FP_SeparateInclusions	int <b>Value:</b> 1493
Constants.FP_ElementSelection	int <b>Value:</b> 1494

<b>Property name</b>	<b>Data Type</b>
Constants.FP_StructuredApplication	int <b>Value:</b> 1495
Constants.FP_CustomElementList	int <b>Value:</b> 1496
Constants.FP_StructuredApplicationList	int <b>Value:</b> 1497
Constants.FP_StructuredApplicationForOpen	int <b>Value:</b> 1498
Constants.FP_UselInitialStructureOfAutoInsertedElements	int <b>Value:</b> 1499
Constants.FP_StructAppAttrConfigFile	int <b>Value:</b> 1500
Constants.FP_ShowElementDescriptiveNames	int <b>Value:</b> 1501
Constants.FP_BkGndColor	int <b>Value:</b> 1502
Constants.FP_TextColor	int <b>Value:</b> 1503
Constants.FP_DitaMode	int <b>Value:</b> 1504
Constants.FP_XSLTTransformationScenarioFile	int <b>Value:</b> 1505
Constants.FP_XSLTProcessors	int <b>Value:</b> 1506
Constants.FV_STRICTLY_VALID	int <b>Value:</b> 0x01
Constants.FV_LOOSELY_VALID	int <b>Value:</b> 0x02
Constants.FV_ALTERNATIVE	int <b>Value:</b> 0x04
Constants.FV_INCLUSION	int <b>Value:</b> 0x08
Constants.FP_HighestLevelElement	int <b>Value:</b> 1509
Constants.FP_ComponentElement	int <b>Value:</b> 1551

## Object Reference

Property name	Data Type
Constants.FP_ExtractElementTags	int <b>Value:</b> 1552
Constants.FP_NextElementDefInDoc	int <b>Value:</b> 1572
Constants.FP_ElementInCatalog	int <b>Value:</b> 1573
Constants.FP_ValidHighestLevel	int <b>Value:</b> 1574
Constants.FP_GeneralRule	int <b>Value:</b> 1575
Constants.FP_Exclusions	int <b>Value:</b> 1576
Constants.FP_Inclusions	int <b>Value:</b> 1577
Constants.FP_GeneralRuleErrorOffsets	int <b>Value:</b> 1578
Constants.FP_ElementDefType	int <b>Value:</b> 1579
Constants.FP_Comment	int <b>Value:</b> 1581
Constants.FP_TextFmtRules	int <b>Value:</b> 1582
Constants.FP_ObjectFmtRules	int <b>Value:</b> 1583
Constants.FP_AttributeDefs	int <b>Value:</b> 1584
Constants.FP_InitStructurePattern	int <b>Value:</b> 1585
Constants.FP_TableTagging	int <b>Value:</b> 1585
Constants.FP_ElementPgffFormat	int <b>Value:</b> 1586
Constants.FP_PrefixRules	int <b>Value:</b> 1587
Constants.FP_SuffixRules	int <b>Value:</b> 1588



<b>Property name</b>	<b>Data Type</b>
Constants.FP_FirstPgfrRules	int <b>Value:</b> 1589
Constants.FP_LastPgfrRules	int <b>Value:</b> 1590
Constants.FP_AlsoInsert	int <b>Value:</b> 1591
Constants.FP_ParsedGeneralRule	int <b>Value:</b> 1592
Constants.FP_ElementDataType	int <b>Value:</b> 1593
Constants.FP_AllowedDataRange	int <b>Value:</b> 1594
Constants.FP_AlsoInserts	int <b>Value:</b> 1595
Constants.FP_DescriptiveTag	int <b>Value:</b> 1596
Constants.FP_ElementDescription	int <b>Value:</b> 1597
Constants.FP_XMLStructuredApplicationList	int <b>Value:</b> 1598
Constants.FP_BannerText	int <b>Value:</b> 1599
Constants.FV_FO_UNSPECIFIED	int <b>Value:</b> 0
Constants.FV_FO_TEXTNODE	int <b>Value:</b> 0
Constants.FV_FO_CONTAINER	int <b>Value:</b> 1
Constants.FV_FO_SYS_VAR	int <b>Value:</b> 2
Constants.FV_FO_XREF	int <b>Value:</b> 3
Constants.FV_FO_MARKER	int <b>Value:</b> 4
Constants.FV_FO_FOOTNOTE	int <b>Value:</b> 5

<b>Property name</b>	<b>Data Type</b>
Constants.FV_FO_GRAPHIC	int <b>Value: 6</b>
Constants.FV_FO_EQN	int <b>Value: 7</b>
Constants.FV_FO_TBL	int <b>Value: 8</b>
Constants.FV_FO_TBL_TITLE	int <b>Value: 9</b>
Constants.FV_FO_TBL_HEADING	int <b>Value: 10</b>
Constants.FV_FO_TBL_BODY	int <b>Value: 11</b>
Constants.FV_FO_TBL_FOOTING	int <b>Value: 12</b>
Constants.FV_FO_TBL_ROW	int <b>Value: 13</b>
Constants.FV_FO_TBL_CELL	int <b>Value: 14</b>
Constants.FV_FO_RUBI_GROUP	int <b>Value: 15</b>
Constants.FV_FO_RUBI	int <b>Value: 16</b>
Constants.FV_FO_NUMTYPES	int <b>Value: 17</b>
Constants.FV_DT_UNSPECIFIED	int <b>Value: 0</b>
Constants.FV_DT_INTEGER	int <b>Value: 1</b>
Constants.FV_DT_FLOAT	int <b>Value: 2</b>
Constants.FV_DT_NUMTYPES	int <b>Value: 3</b>
Constants.FV_AT_STRING	int <b>Value: 0</b>
Constants.FV_AT_STRINGS	int <b>Value: 1</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_AT_CHOICES	int <b>Value: 2</b>
Constants.FV_AT_INTEGER	int <b>Value: 3</b>
Constants.FV_AT_INTEGERS	int <b>Value: 4</b>
Constants.FV_AT_REAL	int <b>Value: 5</b>
Constants.FV_AT_REALS	int <b>Value: 6</b>
Constants.FV_AT_UNIQUE_ID	int <b>Value: 7</b>
Constants.FV_AT_UNIQUE_IDREF	int <b>Value: 8</b>
Constants.FV_AT_UNIQUE_IDREFS	int <b>Value: 9</b>
Constants.FV_AT_NUMTYPES	int <b>Value: 10</b>
Constants.FV_AF_READ_ONLY	int <b>Value: 0x0001</b>
Constants.FV_AF_HIDDEN	int <b>Value: 0x0002</b>
Constants.FV_AF_FIXED	int <b>Value: 0x0004</b>
Constants.FP_FmtRuleType	int <b>Value: 2100</b>
Constants.FP_CountElements	int <b>Value: 2101</b>
Constants.FP_StopCountingAt	int <b>Value: 2102</b>
Constants.FP_FmtRuleClauses	int <b>Value: 2103</b>
Constants.FP_FmtRuleClause	int <b>Value: 2104</b>
Constants.FV_CONTEXT_RULE	int <b>Value: 0</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_LEVEL_RULE	int <b>Value:</b> 1
Constants.FP_Specification	int <b>Value:</b> 2150
Constants.FP_IsTextRange	int <b>Value:</b> 2151
Constants.FP_RuleClauseType	int <b>Value:</b> 2152
Constants.FP_FormatTag	int <b>Value:</b> 2153
Constants.FP_FmtChangeListTag	int <b>Value:</b> 2154
Constants.FP_FmtChangeList	int <b>Value:</b> 2155
Constants.FP_SubFmtRule	int <b>Value:</b> 2156
Constants.FP_ContextLabel	int <b>Value:</b> 2157
Constants.FP_ElemPrefixSuffix	int <b>Value:</b> 2158
Constants.FP_FmtRule	int <b>Value:</b> 2159
Constants.FP_ParsedSpecification	int <b>Value:</b> 2160
Constants.FP_SpecificationForCSS	int <b>Value:</b> 2161
Constants.FP_ContextForCSS	int <b>Value:</b> 2162
Constants.FP_ParseFullSpecification	int <b>Value:</b> 2163
Constants.FV_RC_TAG	int <b>Value:</b> 0
Constants.FV_RC_SUB_FMTRULE	int <b>Value:</b> 1
Constants.FV_RC_CHANGELIST	int <b>Value:</b> 2

<b>Property name</b>	<b>Data Type</b>
Constants.FV_RC_CHANGELIST_TAG	int <b>Value: 3</b>
Constants.FV_EDT_TEXT	int <b>Value: 40</b>
Constants.FV_EDT_TEXTONLY	int <b>Value: 41</b>
Constants.FV_EDT_ANY	int <b>Value: 42</b>
Constants.FV_EDT_ALL	int <b>Value: 43</b>
Constants.FV_EDT_FIRST	int <b>Value: 44</b>
Constants.FV_EDT_MIDDLE	int <b>Value: 45</b>
Constants.FV_EDT_LAST	int <b>Value: 46</b>
Constants.FV_EDT_NOTFIRST	int <b>Value: 47</b>
Constants.FV_EDT_NOTLAST	int <b>Value: 48</b>
Constants.FV_EDT_AFTER	int <b>Value: 49</b>
Constants.FV_EDT_BEFORE	int <b>Value: 50</b>
Constants.FV_EDT_BETWEEN	int <b>Value: 53</b>
Constants.FV_EDT_INSERTPAR	int <b>Value: 23</b>
Constants.FV_EDT_ONLY	int <b>Value: 62</b>
Constants.FV_EDT_ATTRGRPO	int <b>Value: 30</b>
Constants.FV_EDT_ELEMENT	int <b>Value: 25</b>
Constants.FP_ElementDef	int <b>Value: 1621</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_ParentElement	int <b>Value:</b> 1622
Constants.FP_FirstChildElement	int <b>Value:</b> 1623
Constants.FP_LastChildElement	int <b>Value:</b> 1624
Constants.FP_PrevSiblingElement	int <b>Value:</b> 1625
Constants.FP_NextSiblingElement	int <b>Value:</b> 1626
Constants.FP_ElementsCollapsed	int <b>Value:</b> 1627
Constants.FP_BookComponent	int <b>Value:</b> 1628
Constants.FP_Object	int <b>Value:</b> 1630
Constants.FP_MatchingTextClauses	int <b>Value:</b> 1631
Constants.FP_MatchingObjectClauses	int <b>Value:</b> 1632
Constants.FP_Attributes	int <b>Value:</b> 1634
Constants.FP_AttributesEx	int <b>Value:</b> 1670
Constants.FP_AttrDisplay	int <b>Value:</b> 1635
Constants.FP_ElementType	int <b>Value:</b> 1636
Constants.FP_IDAttrValue	int <b>Value:</b> 1637
Constants.FP_MatchingFirstPgfClauses	int <b>Value:</b> 1638
Constants.FP_MatchingLastPgfClauses	int <b>Value:</b> 1639
Constants.FP_MatchingPrefixClauses	int <b>Value:</b> 1640

<b>Property name</b>	<b>Data Type</b>
Constants.FP_MatchingSuffixClauses	int <b>Value:</b> 1641
Constants.FP_MatchesContextInUserString	int <b>Value:</b> 1642
Constants.FP_Namespace	int <b>Value:</b> 1643
Constants.FP_NamespaceScope	int <b>Value:</b> 1644
Constants.FP_NumNamespaces	int <b>Value:</b> 1645
Constants.FP_NextElementDFS	int <b>Value:</b> 2334
Constants.FP_PrevElementDFS	int <b>Value:</b> 2335
Constants.FP_ElementsTopicRef	int <b>Value:</b> 2336
Constants.FP_ElementsTopicHead	int <b>Value:</b> 2337
Constants.FP_ElementsTopicGroup	int <b>Value:</b> 2338
Constants.FP_TopicRefAbsolutePath	int <b>Value:</b> 2347
Constants.FP_Element	int <b>Value:</b> 2170
Constants.FP_TblElement	int <b>Value:</b> 2171
Constants.FP_TblTitleElement	int <b>Value:</b> 2172
Constants.FP_TblHeaderElement	int <b>Value:</b> 2173
Constants.FP_TblBodyElement	int <b>Value:</b> 2174
Constants.FP_TblFooterElement	int <b>Value:</b> 2175
Constants.FP_RubiElement	int <b>Value:</b> 2176

<b>Property name</b>	<b>Data Type</b>
Constants.FP_AllowAsSpecialCase	int <b>Value:</b> 1649
Constants.FP_ElementsUndefined	int <b>Value:</b> 1650
Constants.FP_ContentsStrictlyValid	int <b>Value:</b> 1651
Constants.FP_ContentsLooselyValid	int <b>Value:</b> 1652
Constants.FP_ContentNeededAtBegin	int <b>Value:</b> 1653
Constants.FP_HoleAtEnd	int <b>Value:</b> 1654
Constants.FP_ContentNeededAtEnd	int <b>Value:</b> 1654
Constants.FP_ElementsExcludedInContext	int <b>Value:</b> 1655
Constants.FP_ElementsInvalidInParent	int <b>Value:</b> 1656
Constants.FP_ElementsInvalidInPosition	int <b>Value:</b> 1657
Constants.FP_ElementTypeMismatch	int <b>Value:</b> 1658
Constants.FP_HoleBeforeElement	int <b>Value:</b> 1659
Constants.FP_TextsInvalidInElement	int <b>Value:</b> 1660
Constants.FP_InvalidHighestLevel	int <b>Value:</b> 1661
Constants.FP_BookComponentMissing	int <b>Value:</b> 1662
Constants.FP_ErrorInBookComponent	int <b>Value:</b> 1663
Constants.FP_NextInvalidElement	int <b>Value:</b> 1664
Constants.FP_ContentMustBeEmpty	int <b>Value:</b> 1665



<b>Property name</b>	<b>Data Type</b>
Constants.FP_AttributeValueRequired	int <b>Value:</b> 1666
Constants.FP_AttributeValueInvalid	int <b>Value:</b> 1667
Constants.FP_ValidationFlags	int <b>Value:</b> 1668
Constants.FP_ElementMarkedForNamedDestination	int <b>Value:</b> 1669
Constants.FV_ELEM_UNDEFINED	int <b>Value:</b> 0x0001
Constants.FV_ELEM_TYPE_MISMATCH	int <b>Value:</b> 0x0002
Constants.FV_ELEM_EXCLUDED	int <b>Value:</b> 0x0004
Constants.FV_ELEM_INVALID_IN_PARENT	int <b>Value:</b> 0x0008
Constants.FV_ELEM_INVALID_AT_POSITION	int <b>Value:</b> 0x0010
Constants.FV_ELEM_HAS_TEXT_INVALID	int <b>Value:</b> 0x0020
Constants.FV_ELEM_CONTENT_MUST_BE_EMPTY	int <b>Value:</b> 0x0040
Constants.FV_ELEM_MISSING_CONTENT_BEFORE	int <b>Value:</b> 0x0080
Constants.FV_ELEM_MISSING_CONTENT_AT_BEG	int <b>Value:</b> 0x0100
Constants.FV_ELEM_MISSING_CONTENT_AT_END	int <b>Value:</b> 0x0200
Constants.FV_ELEM_NOT_VALID_AS_ROOT	int <b>Value:</b> 0x0400
Constants.FV_ELEM_BOOK_COMP_MISSING	int <b>Value:</b> 0x0800
Constants.FV_ELEM_BOOK_COMP_INVALID	int <b>Value:</b> 0x1000
Constants.FV_ELEM_ATTRVAL_REQUIRED	int <b>Value:</b> 0x2000

## Object Reference

Property name	Data Type
Constants.FV_ELEM_ATTRVAL_INVALID	int <b>Value:</b> 0x4000
Constants.FV_ELEM_CONTENT_STRICTLY_VALID	int <b>Value:</b> 0x10000
Constants.FV_ELEM_CONTENT_LOOSELY_VALID	int <b>Value:</b> 0x20000
Constants.FV_ELEM_INVISIBLE_ATTRVAL_REQUIRED	int <b>Value:</b> 0x40000
Constants.FV_ELEM_INVISIBLE_ATTRVAL_INVALID	int <b>Value:</b> 0x80000
Constants.FV_AV_REQUIRED	int <b>Value:</b> 1
Constants.FV_AV_INVALID_CHOICE	int <b>Value:</b> 2
Constants.FV_AV_INVALID_FORMAT	int <b>Value:</b> 3
Constants.FV_AV_IDREF_UNRESOLVED	int <b>Value:</b> 4
Constants.FV_AV_ID_DUPLICATE_IN_DOC	int <b>Value:</b> 5
Constants.FV_AV_ID_DUPLICATE_IN_BOOK	int <b>Value:</b> 6
Constants.FV_AV_TOO_MANY_TOKENS	int <b>Value:</b> 7
Constants.FV_AV_UNDEFINED	int <b>Value:</b> 8
Constants.FV_AV_OUT_OF_RANGE	int <b>Value:</b> 9
Constants.FV_OVERRIDDEN_DEFAULT	int <b>Value:</b> 0x00
Constants.FV_OVERRIDDEN_NOVALUE	int <b>Value:</b> 0x01
Constants.FV_OVERRIDDEN_CONREF_DUCT	int <b>Value:</b> 0x02
Constants.FV_OVERRIDDEN_KEYDEF_TO_KEYREF	int <b>Value:</b> 0x03

<b>Property name</b>	<b>Data Type</b>
Constants.FV_Element	int <b>Value:</b> 0
Constants.FV_ElementsOfType	int <b>Value:</b> 1
Constants.FV_AllElements	int <b>Value:</b> 2
Constants.FP_NextFmtChangeListInDoc	int <b>Value:</b> 2000
Constants.FP_FmtChangeListInCatalog	int <b>Value:</b> 2021
Constants.FP_PgfCatalogReference	int <b>Value:</b> 2001
Constants.FP_SpaceAboveChange	int <b>Value:</b> 2002
Constants.FP_SpaceBelowChange	int <b>Value:</b> 2003
Constants.FP_LeftIndentChange	int <b>Value:</b> 2004
Constants.FP_FirstIndentChange	int <b>Value:</b> 2005
Constants.FP_RightIndentChange	int <b>Value:</b> 2006
Constants.FP_LeadingChange	int <b>Value:</b> 2007
Constants.FP_LineSpacingFixed	int <b>Value:</b> 2008
Constants.FP_TopSepAtIndent	int <b>Value:</b> 2009
Constants.FP_BottomSepAtIndent	int <b>Value:</b> 2010
Constants.FP_FontSizeChange	int <b>Value:</b> 2012
Constants.FP_SpreadChange	int <b>Value:</b> 2013
Constants.FP_MoveTabs	int <b>Value:</b> 2014

<b>Property name</b>	<b>Data Type</b>
Constants.FP_CellTopMarginFixed	int <b>Value:</b> 2015
Constants.FP_CellBottomMarginFixed	int <b>Value:</b> 2016
Constants.FP_CellLeftMarginFixed	int <b>Value:</b> 2017
Constants.FP_CellRightMarginFixed	int <b>Value:</b> 2018
Constants.FP_FirstIndentsRelative	int <b>Value:</b> 2019
Constants.FP_FirstIndentRelPos	int <b>Value:</b> 2020
Constants.FP_CellTopMarginChange	int <b>Value:</b> 2022
Constants.FP_CellBottomMarginChange	int <b>Value:</b> 2023
Constants.FP_CellLeftMarginChange	int <b>Value:</b> 2024
Constants.FP_CellRightMarginChange	int <b>Value:</b> 2025
Constants.FP_StretchChange	int <b>Value:</b> 2026
Constants.FF_IED_REMOVE_OVERRIDES	int <b>Value:</b> 0x0001
Constants.FF_IED_REMOVE_BOOK_INFO	int <b>Value:</b> 0x0002
Constants.FF_IED_DO_NOT_IMPORT_EDD	int <b>Value:</b> 0x0004
Constants.FF_IED_NO_NOTIFY	int <b>Value:</b> 0x0008
Constants.FF_IED_DELETE_EMPTY_PAGES	int <b>Value:</b> 0x0010
Constants.FF_IED_DO_NOT_IMPORT_STRUCTAPP	int <b>Value:</b> 0x0040
Constants.FP_Text	int <b>Value:</b> 1701

<b>Property name</b>	<b>Data Type</b>
Constants.FP_State	int <b>Value:</b> 1702
Constants.FV_DlgOptNotActive	int <b>Value:</b> 0
Constants.FV_DlgOptActive	int <b>Value:</b> 1
Constants.FV_DlgOptDontCare	int <b>Value:</b> 2
Constants.FP_Labels	int <b>Value:</b> 1703
Constants.FP_NumLines	int <b>Value:</b> 1704
Constants.FP_FirstVis	int <b>Value:</b> 1705
Constants.FP_DoubleClick	int <b>Value:</b> 1706
Constants.FP_NumItems	int <b>Value:</b> 1707
Constants.FP_Sensitivity	int <b>Value:</b> 1708
Constants.FP_MinVal	int <b>Value:</b> 1709
Constants.FP_MaxVal	int <b>Value:</b> 1710
Constants.FP_IncrVal	int <b>Value:</b> 1711
Constants.FP_Size	int <b>Value:</b> 1712
Constants.FP_Visibility	int <b>Value:</b> 1713
Constants.FP_PasswordStyle	int <b>Value:</b> 1714
Constants.FP_ReadOnly	int <b>Value:</b> 1723
Constants.FP_TabStops	int <b>Value:</b> 1715

<b>Property name</b>	<b>Data Type</b>
Constants.FP_Icon	int <b>Value:</b> 1716
Constants.FP_WindowHandle	int <b>Value:</b> 1717
Constants.FP_SortingEnabled	int <b>Value:</b> 1718
Constants.FP_SelectedRowItems	int <b>Value:</b> 1719
Constants.FP_LabelColor	int <b>Value:</b> 1720
Constants.FP_LabelBold	int <b>Value:</b> 1721
Constants.FP_Box_BG_Color	int <b>Value:</b> 1722
Constants.FV_DlgClose	int <b>Value:</b> -1
Constants.FV_DlgReset	int <b>Value:</b> -2
Constants.FV_DlgNoChange	int <b>Value:</b> -3
Constants.FV_DlgEnter	int <b>Value:</b> -4
Constants.FV_DlgUndo	int <b>Value:</b> -5
Constants.FV_DlgPrevPage	int <b>Value:</b> -6
Constants.FV_DlgNextPage	int <b>Value:</b> -7
Constants.FV_DlgNeedsUpdate	int <b>Value:</b> -8
Constants.FV_DlgInit	int <b>Value:</b> -9
Constants.FV_DlgHide	int <b>Value:</b> -10
Constants.FV_DlgShow	int <b>Value:</b> -11

<b>Property name</b>	<b>Data Type</b>
Constants.FV_DlgResize	int <b>Value:</b> -12
Constants.FV_DlgPrefSetInitialState	int <b>Value:</b> -100
Constants.FV_DlgPrefVerify	int <b>Value:</b> -101
Constants.FV_DlgPrefCommit	int <b>Value:</b> -102
Constants.FV_EvShift	int <b>Value:</b> 0x0001
Constants.FV_EvControl	int <b>Value:</b> 0x0002
Constants.FV_EvMeta	int <b>Value:</b> 0x0004
Constants.FV_EvOption	int <b>Value:</b> 0x0004
Constants.FV_EvCaps	int <b>Value:</b> 0x0008
Constants.FV_EvCommand	int <b>Value:</b> 0x0010
Constants.FV_EvMax	int <b>Value:</b> 0x0010
Constants.FP_DbItemNum	int <b>Value:</b> 1800
Constants.FP_DbType	int <b>Value:</b> 1801
Constants.FP_DbIdentifier	int <b>Value:</b> 1802
Constants.FP_DbVarLabelWidth	int <b>Value:</b> 1803
Constants.FP_DbStuffItem	int <b>Value:</b> 1804
Constants.FP_DbSbxNumLines	int <b>Value:</b> 1805
Constants.FP_DbRadioButtonGroup	int <b>Value:</b> 1806

<b>Property name</b>	<b>Data Type</b>
Constants.FP_DbCheckBoxState	int <b>Value:</b> 1807
Constants.FP_DbFirstFocus	int <b>Value:</b> 1808
Constants.FP_DbDefaultButton	int <b>Value:</b> 1809
Constants.FP_DbOKButton	int <b>Value:</b> 1810
Constants.FP_DbCancelButton	int <b>Value:</b> 1811
Constants.FP_DbFbTextBox	int <b>Value:</b> 1812
Constants.FP_DbFbScrollBar	int <b>Value:</b> 1813
Constants.FP_DbFbCurrentDir	int <b>Value:</b> 1814
Constants.FP_DbFbStatus	int <b>Value:</b> 1815
Constants.FP_DbTitleLabel	int <b>Value:</b> 1816
Constants.FP_DbAttributes	int <b>Value:</b> 1817
Constants.FP_DbEditable	int <b>Value:</b> 1818
Constants.FP_DbNoHelp	int <b>Value:</b> 1819
Constants.FP_LineAscent	int <b>Value:</b> 1900
Constants.FP_LineDescent	int <b>Value:</b> 1901
Constants.FP_LineBaseline	int <b>Value:</b> 1902
Constants.FP_MenuItemsEnabled	int <b>Value:</b> 1922
Constants.FP_NextMenuItemInMenu	int <b>Value:</b> 1923



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

<b>Property name</b>	<b>Data Type</b>
Constants.FP_ExpandOMaticParent	int <b>Value:</b> 1934
Constants.FP_EnabledWhen	int <b>Value:</b> 1935
Constants.FV_ENABLE_ALWAYS_ENABLE	int <b>Value:</b> 1
Constants.FV_ENABLE_ALWAYS_DISABLE	int <b>Value:</b> 2
Constants.FV_ENABLE_IN_PARA_TEXT	int <b>Value:</b> 3
Constants.FV_ENABLE_IN_TEXT_LINE	int <b>Value:</b> 4
Constants.FV_ENABLE_IS_TEXT_SEL	int <b>Value:</b> 5
Constants.FV_ENABLE_IN_MATH	int <b>Value:</b> 6
Constants.FV_ENABLE_IN_TEXT	int <b>Value:</b> 7
Constants.FV_ENABLE_OBJ_PROPS	int <b>Value:</b> 8
Constants.FV_ENABLE_IN_TABLE	int <b>Value:</b> 9
Constants.FV_ENABLE_IN_TABLE_TITLE	int <b>Value:</b> 10
Constants.FV_ENABLE_IN_CELL_TEXT	int <b>Value:</b> 11
Constants.FV_ENABLE_IS_CELL	int <b>Value:</b> 12
Constants.FV_ENABLE_IS_CELLS	int <b>Value:</b> 13
Constants.FV_ENABLE_IS_TABLE	int <b>Value:</b> 14
Constants.FV_ENABLE_IS_OBJ	int <b>Value:</b> 15
Constants.FV_ENABLE_IS_TEXT_FRAME	int <b>Value:</b> 16

<b>Property name</b>	<b>Data Type</b>
Constants.FV_ENABLE_IS_OR_IN_FRAME	int <b>Value: 17</b>
Constants.FV_ENABLE_IS_AFRAME	int <b>Value: 18</b>
Constants.FV_ENABLE_IS_TEXT_INSET	int <b>Value: 19</b>
Constants.FV_ENABLE_IS_GRAPHIC_INSET	int <b>Value: 20</b>
Constants.FV_ENABLE_IN_FLOW	int <b>Value: 21</b>
Constants.FV_ENABLE_COPY	int <b>Value: 22</b>
Constants.FV_ENABLE_COPY_FONT	int <b>Value: 23</b>
Constants.FV_ENABLE_CAN_PASTE	int <b>Value: 24</b>
Constants.FV_ENABLE_IS_VIEW_ONLY	int <b>Value: 25</b>
Constants.FV_ENABLE_NEEDS_DOCP_ONLY	int <b>Value: 26</b>
Constants.FV_ENABLE_NEEDS_BOOKP_ONLY	int <b>Value: 27</b>
Constants.FV_ENABLE_NEEDS_DOCP_OR_BOOKP	int <b>Value: 28</b>
Constants.FV_ENABLE_BOOK_HAS_SELECTION	int <b>Value: 29</b>
Constants.FV_ENABLE_DOC_OR_BOOK_HAS_SELECTION	int <b>Value: 30</b>
Constants.FP_Fcode	int <b>Value: 1936</b>
Constants.FP_Fcodes	int <b>Value: 1937</b>
Constants.FP_HelpLink	int <b>Value: 1938</b>
Constants.FP_HasShiftOrUnshiftCommand	int <b>Value: 1939</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_ITEM_HAS_SHIFT_COMMAND	int <b>Value: 1</b>
Constants.FV_ITEM_HAS_UNSHIFT_COMMAND	int <b>Value: 2</b>
Constants.FV_ITEM_HAS_NO_SHIFT_OR_UNSHIFT_COMMAND	int <b>Value: 3</b>
Constants.FP_ShiftOrUnshiftCommand	int <b>Value: 1940</b>
Constants.FP_Mode	int <b>Value: 1941</b>
Constants.FV_MODE_MATH	int <b>Value: 1</b>
Constants.FV_MODE_NONMATH	int <b>Value: 2</b>
Constants.FV_MODE_ALL	int <b>Value: 3</b>
Constants.FP_NextCommandInSession	int <b>Value: 1942</b>
Constants.FP_Hypertext	int <b>Value: 1943</b>
Constants.FP_NextTiInDoc	int <b>Value: 2050</b>
Constants.FP_TiAutomaticUpdate	int <b>Value: 2051</b>
Constants.FV_UpdateAllClientTi	int <b>Value: 1</b>
Constants.FV_UpdateAllAutomaticClientTi	int <b>Value: 2</b>
Constants.FV_UpdateAllManualClientTi	int <b>Value: 3</b>
Constants.FP_LastUpdate	int <b>Value: 2052</b>
Constants.FP_TiFile	int <b>Value: 2053</b>
Constants.FP_TiMacEdition	int <b>Value: 2054</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_ImportHint	int <b>Value:</b> 2055
Constants.FP_TiLocked	int <b>Value:</b> 2056
Constants.FP_TiFileModDate	int <b>Value:</b> 2057
Constants.FP_TiIsNested	int <b>Value:</b> 2058
Constants.FP_TiMainFlow	int <b>Value:</b> 2059
Constants.FP_TiFlowName	int <b>Value:</b> 2060
Constants.FP_TiFlowPageSpace	int <b>Value:</b> 2061
Constants.FV_BODY_PAGE	int <b>Value:</b> 0x00
Constants.FV_REFERENCE_PAGE	int <b>Value:</b> 0x02
Constants.FP_TiFormat	int <b>Value:</b> 2062
Constants.FP_TiRemovePageBreaks	int <b>Value:</b> 2063
Constants.FP_TiRemoveOverrides	int <b>Value:</b> 2064
Constants.FP_TiTextEncoding	int <b>Value:</b> 2073
Constants.FV_IsoLatin	int <b>Value:</b> 1
Constants.FV_ASCII	int <b>Value:</b> 2
Constants.FV_ANSI	int <b>Value:</b> 3
Constants.FV_MacANSI	int <b>Value:</b> 4
Constants.FV_JIS	int <b>Value:</b> 5

<b>Property name</b>	<b>Data Type</b>
Constants.FV_Shif_JIS	int <b>Value: 6</b>
Constants.FV_EUC	int <b>Value: 7</b>
Constants.FV_BIG5	int <b>Value: 8</b>
Constants.FV_EUC_CNS	int <b>Value: 9</b>
Constants.FV_GB	int <b>Value: 10</b>
Constants.FV_HZ	int <b>Value: 11</b>
Constants.FV_Korean	int <b>Value: 12</b>
Constants.FP_TiEOLisEOP	int <b>Value: 2065</b>
Constants.FP_TiTbITag	int <b>Value: 2066</b>
Constants.FP_TiNumHeaderRows	int <b>Value: 2067</b>
Constants.FP_TiHeadersEmpty	int <b>Value: 2068</b>
Constants.FP_TiByRows	int <b>Value: 2069</b>
Constants.FP_TiSeparator	int <b>Value: 2070</b>
Constants.FP_TiNumSeparators	int <b>Value: 2071</b>
Constants.FP_TiNumCols	int <b>Value: 2072</b>
Constants.FP_TiIsUnresolved	int <b>Value: 2075</b>
Constants.FP_TiClientName	int <b>Value: 2076</b>
Constants.FP_TiClientSource	int <b>Value: 2077</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_TiClientType	int <b>Value:</b> 2078
Constants.FP_TiClientData	int <b>Value:</b> 2079
Constants.FP_TiClientUpdateSilent	int <b>Value:</b> 2080
Constants.FP_ContentHeight	int <b>Value:</b> 2219
Constants.FP_ParentTextFrame	int <b>Value:</b> 2200
Constants.FP_PrevSubCol	int <b>Value:</b> 2201
Constants.FP_NextSubCol	int <b>Value:</b> 2202
Constants.FP_Overflowed	int <b>Value:</b> 983
Constants.FP_FirstCombinedFontDefnInDoc	int <b>Value:</b> 2280
Constants.FP_NextCombinedFontDefnInDoc	int <b>Value:</b> 2281
Constants.FP_BaseFamily	int <b>Value:</b> 2282
Constants.FP_AllowBaseFamilyBoldedAndObliqued	int <b>Value:</b> 2283
Constants.FP_WesternFamily	int <b>Value:</b> 2284
Constants.FP_WesternSize	int <b>Value:</b> 2285
Constants.FP_WesternShift	int <b>Value:</b> 2286
Constants.FP_CombinedFontFamily	int <b>Value:</b> 2287
Constants.FS_FindText	int <b>Value:</b> 1
Constants.FS_FindElementTag	int <b>Value:</b> 2

<b>Property name</b>	<b>Data Type</b>
Constants.FV_FindElemTag	int <b>Value: 0</b>
Constants.FV_FindAttrName	int <b>Value: 1</b>
Constants.FV_FindAttrValue	int <b>Value: 2</b>
Constants.FV_NumFindElementItems	int <b>Value: 2+1</b>
Constants.FS_FindCharFmt	int <b>Value: 3</b>
Constants.FS_FindPgftag	int <b>Value: 4</b>
Constants.FS_FindCharTag	int <b>Value: 5</b>
Constants.FS_FindTableTag	int <b>Value: 6</b>
Constants.FS_FindObject	int <b>Value: 8</b>
Constants.FV_FindAnyMarker	int <b>Value: 0</b>
Constants.FV_FindAnyXRef	int <b>Value: 1</b>
Constants.FV_FindUnresolvedXRef	int <b>Value: 2</b>
Constants.FV_FindAnyTextInset	int <b>Value: 3</b>
Constants.FV_FindUnresolvedTextInset	int <b>Value: 4</b>
Constants.FV_FindAnyPub	int <b>Value: 5</b>
Constants.FV_FindAnyVariable	int <b>Value: 6</b>
Constants.FV_FindAnchoredFrame	int <b>Value: 7</b>
Constants.FV_FindFootnote	int <b>Value: 8</b>



<b>Property name</b>	<b>Data Type</b>
Constants.FV_FindAnyTable	int <b>Value: 9</b>
Constants.FV_FindAutomaticHyphen	int <b>Value: 10</b>
Constants.FV_FindAnyRubi	int <b>Value: 11</b>
Constants.FV_FindPgFormatOverride	int <b>Value: 12</b>
Constants.FV_FindCharacterFormatOverride	int <b>Value: 13</b>
Constants.FV_FindTableFormatOverride	int <b>Value: 14</b>
Constants.FS_FindMarkerOfType	int <b>Value: 9</b>
Constants.FS_FindMarkerText	int <b>Value: 10</b>
Constants.FS_FindXRefWithFormat	int <b>Value: 11</b>
Constants.FS_FindNamedVariable	int <b>Value: 12</b>
Constants.FS_FindCondTextInCondTags	int <b>Value: 13</b>
Constants.FS_FindCondTextNotInCondTags	int <b>Value: 14</b>
Constants.FS_FindCustomizationFlags	int <b>Value: 15</b>
Constants.FS_FindWrap	int <b>Value: 16</b>
Constants.FS_SpellOptions	int <b>Value: 1000</b>
Constants.FS_TwoInARowString	int <b>Value: 1001</b>
Constants.FS_TextBeforeString	int <b>Value: 1002</b>
Constants.FS_TextAfterString	int <b>Value: 1003</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FS_WordContainString	int <b>Value:</b> 1004
Constants.FS_SpellAction	int <b>Value:</b> 1005
Constants.FV_CheckDocument	int <b>Value:</b> 1
Constants.FV_CheckCurrentPage	int <b>Value:</b> 2
Constants.FV_WriteUnknownWordsToFile	int <b>Value:</b> 3
Constants.FS_OutputFilePathName	int <b>Value:</b> 1006
Constants.FS_SpellWrap	int <b>Value:</b> 1007
Constants.FS_NumSpellParams	int <b>Value:</b> (1007-1000+1)
Constants.FV_SpellMisspelling	int <b>Value:</b> 1
Constants.FV_SpellCapitalization	int <b>Value:</b> 2
Constants.FV_SpellRepeatedWord	int <b>Value:</b> 3
Constants.FV_SpellRepeatedLetter	int <b>Value:</b> 4
Constants.FV_SpellHyphenation	int <b>Value:</b> 5
Constants.FV_SpellExtraSpace	int <b>Value:</b> 6
Constants.FV_SpellSpaceAfter	int <b>Value:</b> 7
Constants.FV_SpellSpaceBefore	int <b>Value:</b> 8
Constants.FV_SpellStraightQuotes	int <b>Value:</b> 9
Constants.FV_UserDictionary	int <b>Value:</b> 1

<b>Property name</b>	<b>Data Type</b>
Constants.FV_DocumentDictionary	int <b>Value: 2</b>
Constants.FV_WriteDictionaryToFile	int <b>Value: 4</b>
Constants.FV_MergeDictionaryContents	int <b>Value: 5</b>
Constants.FV_SetDictionaryToNoneDictionary	int <b>Value: 6</b>
Constants.FV_ClearDocDictionary	int <b>Value: 7</b>
Constants.FV_ChangePersonalDictionary	int <b>Value: 8</b>
Constants.FV_GetDictionaryStrings	int <b>Value: 9</b>
Constants.FP_CursorData	int <b>Value: 2294</b>
Constants.FP_CursorTypes	int <b>Value: 2295</b>
Constants.FP_StructureCursorTypes	int <b>Value: 2296</b>
Constants.FP_SecNumComputeMethod	int <b>Value: 915</b>
Constants.FP_SectionNumStyle	int <b>Value: 916</b>
Constants.FP_SectionNumber	int <b>Value: 917</b>
Constants.FP_SectionNumText	int <b>Value: 918</b>
Constants.FP_SubsecNumComputeMethod	int <b>Value: 815</b>
Constants.FP_SubsectionNumStyle	int <b>Value: 816</b>
Constants.FP_SubsectionNumber	int <b>Value: 817</b>
Constants.FP_SubsectionNumText	int <b>Value: 818</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_DoNotGenerateErrorLog	int <b>Value:</b> 819
Constants.FP_DocOpenClientEncounteredErrors	int <b>Value:</b> 820
Constants.FP_OpenAndSaveXmlBookComponentDoc	int <b>Value:</b> 822
Constants.FP_XmlIsBook	int <b>Value:</b> 823
Constants.FP_UseFMConsoleForErrorLog	int <b>Value:</b> 825
Constants.FP_FMConsoleString	int <b>Value:</b> 826
Constants.FP_IsTempOpenSave	int <b>Value:</b> 827
Constants.FP_AllowNewFileURL	int <b>Value:</b> 2416
Constants.FP_DoPostXSLTValidationOnExport	int <b>Value:</b> 2417
Constants.FP_DoNotExportInvalidXML	int <b>Value:</b> 2418
Constants.FP_SuppressXMLParserWarnings	int <b>Value:</b> 2419
Constants.FP_RemoveExtraWhiteSpacesOnXMLImport	int <b>Value:</b> 2830
Constants.FP_NoMultiMediaInPDF	int <b>Value:</b> 2831
Constants.FP_InsetGenericData	int <b>Value:</b> 2832
Constants.FP_InTableSortOperation	int <b>Value:</b> 2833
Constants.FP_InsetGfxName	int <b>Value:</b> 2834
Constants.FP_InsetGfxPlayWindowInPdf	int <b>Value:</b> 2835
Constants.FP_InsetGfxActiveInPdf	int <b>Value:</b> 2836

<b>Property name</b>	<b>Data Type</b>
Constants.FP_InsetJavaScriptAttached	int <b>Value:</b> 2837
Constants.FP_InsetJavaScriptFile	int <b>Value:</b> 2838
Constants.FP_InsetMonikerPath	int <b>Value:</b> 2848
Constants.FP_InsetMonikerFilePath	int <b>Value:</b> 2849
Constants.FP_IsFateRun	int <b>Value:</b> 2847
Constants.FP_DefaultVectorFormatForXMLExport	int <b>Value:</b> 2874
Constants.FV_CmdOpenXmlElementId	int <b>Value:</b> 821
Constants.FV_FileTypeBinaryDoc	int <b>Value:</b> 1
Constants.FV_FileTypeBinaryBook	int <b>Value:</b> 2
Constants.FV_FileTypeMif	int <b>Value:</b> 3
Constants.FV_FileTypeXml	int <b>Value:</b> 4
Constants.FV_FileTypeSgml	int <b>Value:</b> 5
Constants.FV_FileTypeText	int <b>Value:</b> 6
Constants.FV_FileTypeFilter	int <b>Value:</b> 7
Constants.FV_FileTypeFolder	int <b>Value:</b> 8
Constants.FV_FileTypeGeneral	int <b>Value:</b> 9
Constants.FV_FileTypeErrorDoc	int <b>Value:</b> 10
Constants.FV_FileTypeEquation	int <b>Value:</b> 11

<b>Property name</b>	<b>Data Type</b>
Constants.FV_FileTypeThesaurus	int <b>Value: 12</b>
Constants.FV_FileTypeCompositeDoc	int <b>Value: 13</b>
Constants.FV_FileTypeSummaryDoc	int <b>Value: 14</b>
Constants.FS_StructuredApplication	int <b>Value: 1</b>
Constants.FS_Doctype	int <b>Value: 2</b>
Constants.FS_PublicId	int <b>Value: 3</b>
Constants.FS_SystemId	int <b>Value: 4</b>
Constants.FS_Extension	int <b>Value: 5</b>
Constants.FS_Visible	int <b>Value: 6</b>
Constants.FS_NumNewXMLParams	int <b>Value: 6</b>
Constants.FV_ListViewInit	int <b>Value: 1</b>
Constants.FV_ListViewAddColumn	int <b>Value: 2</b>
Constants.FV_ListViewAddRow	int <b>Value: 3</b>
Constants.FV_ListViewDeleteAllRows	int <b>Value: 4</b>
Constants.FV_ListViewReSortAllRows	int <b>Value: 5</b>
Constants.FV_ListViewActionMax	int <b>Value: 6</b>
Constants.FV_ListViewInitNumParams	int <b>Value: 0</b>
Constants.FV_ListViewAddColumnNumParams	int <b>Value: 4</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_ListViewAddColumnParam1	int <b>Value: 3</b>
Constants.FV_ListViewAddColumnParam2	int <b>Value: 1</b>
Constants.FV_ListViewAddColumnParam3	int <b>Value: 1</b>
Constants.FV_ListViewAddColumnParam4	int <b>Value: 1</b>
Constants.FV_ListViewAddRowNumParams	int <b>Value: 1</b>
Constants.FV_ListViewAddRowParam1	int <b>Value: 6</b>
Constants.FV_ListViewDeleteAllRowsNumParams	int <b>Value: 0</b>
Constants.FV_ListViewReSortAllRowsNumParams	int <b>Value: 0</b>
Constants.FP_LineNumDistance	int <b>Value: 4001</b>
Constants.FP_LineNumRestart	int <b>Value: 4002</b>
Constants.FP_LineNumShow	int <b>Value: 4003</b>
Constants.FP_LineNumFont	int <b>Value: 4004</b>
Constants.FP_LineNumSize	int <b>Value: 4005</b>
Constants.FP_LineNumColor	int <b>Value: 4006</b>
Constants.FP_IsDefault	int <b>Value: 2850</b>
Constants.FP_IsStale	int <b>Value: 2851</b>
Constants.FP_NextKeyCatalogInSession	int <b>Value: 2852</b>
Constants.FP_NotLoaded	int <b>Value: 2853</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FP_Source	int <b>Value:</b> 2854
Constants.FP_SourceType	int <b>Value:</b> 2855
Constants.FP_KeyCount	int <b>Value:</b> 2860
Constants.FP_KeyCatalogClientName	int <b>Value:</b> 2873
Constants.FV_KeySrcTypeNone	int <b>Value:</b> 0
Constants.FV_KeySrcTypeDitamap	int <b>Value:</b> 1
Constants.FV_DocKeyCatalogTypeDefault	int <b>Value:</b> 0
Constants.FV_DocKeyCatalogTypeSpecified	int <b>Value:</b> 1
Constants.FV_DocKeyCatalogTypeNone	int <b>Value:</b> 2
Constants.FV_DocKeyCatalogTypeMax	int <b>Value:</b> 3
Constants.FV_KeydefVariableInfoLen	int <b>Value:</b> 3
Constants.FV_KeydefVariableInfoTagPrimary	int <b>Value:</b> 0
Constants.FV_KeydefVariableInfoTagSecondary	int <b>Value:</b> 1
Constants.FV_KeydefVariableInfoText	int <b>Value:</b> 2
Constants.FV_KeyDefFieldsTypePrimary	int <b>Value:</b> 0
Constants.FV_KeyDefFieldsTypeAll	int <b>Value:</b> 1
Constants.FV_KeyDefFieldsTypeMax	int <b>Value:</b> 2
Constants.FV_KeydefKeyTag	int <b>Value:</b> 1



<b>Property name</b>	<b>Data Type</b>
Constants.FV_KeydefKeyTarget	int <b>Value: 2</b>
Constants.FV_KeydefKeySrcFile	int <b>Value: 3</b>
Constants.FV_KeydefKeyDuplicate	int <b>Value: 4</b>
Constants.FV_KeyDefMaxPrimaryFields	int <b>Value: 4+1</b>
Constants.FV_KeydefKeySrcType	int <b>Value: 5</b>
Constants.FV_KeydefKeyVarList	int <b>Value: 6</b>
Constants.FV_KeydefKeyDefaultText	int <b>Value: 7</b>
Constants.FV_KeydefKeyFoundInRefFile	int <b>Value: 8</b>
Constants.FV_KeydefKeyInValid	int <b>Value: 9</b>
Constants.FV_KeydefKeyAttrs	int <b>Value: 10</b>
Constants.FV_KeyDefMaxFields	int <b>Value: 11</b>
Constants.FV_KeydefKeyTagType	int <b>Value: 3</b>
Constants.FV_KeydefKeyTargetType	int <b>Value: 3</b>
Constants.FV_KeydefKeySrcFileType	int <b>Value: 3</b>
Constants.FV_KeydefKeyDuplicateType	int <b>Value: 1</b>
Constants.FV_KeydefKeySrcTypeType	int <b>Value: 1</b>
Constants.FV_KeydefKeyVarListType	int <b>Value: 20</b>
Constants.FV_KeydefKeyDefaultTextType	int <b>Value: 3</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FV_KeydefKeyFoundInRefFileType	int <b>Value: 1</b>
Constants.FV_KeydefKeyInValidType	int <b>Value: 1</b>
Constants.FV_KeydefKeyAttrsType	int <b>Value: 21</b>
Constants.FV_KeyDefFieldSize	int <b>Value: 2</b>
Constants.FF_DUPLICATE_KEY_DEFINITION	int <b>Value: 0x01</b>
Constants.FF_FOUND_IN_REFERENCED_FILE	int <b>Value: 0x02</b>
Constants.FF_INVALID_KEY	int <b>Value: 0x04</b>
Constants.FV_DITAObjTypeAuto	int <b>Value: 0</b>
Constants.FV_DITAObjTypeConref	int <b>Value: 1</b>
Constants.FV_DITAObjTypeXref	int <b>Value: 2</b>
Constants.FV_DITAObjTypeLink	int <b>Value: 3</b>
Constants.FV_DITAObjTypeTopicref	int <b>Value: 4</b>
Constants.FV_DITAObjTypeTopicsetref	int <b>Value: 5</b>
Constants.FV_DITAObjTypeMax	int <b>Value: 6</b>
Constants.FF_DITAUpdateAllConrefs	int <b>Value: 0x01</b>
Constants.FF_DITAUpdateAllXrefs	int <b>Value: 0x02</b>
Constants.FF_DITAUpdateAllLinks	int <b>Value: 0x04</b>
Constants.FF_DITAUpdateAllTopicrefs	int <b>Value: 0x08</b>

<b>Property name</b>	<b>Data Type</b>
Constants.FF_DITAUpdateAllTopicsetrefs	int <b>Value:</b> 0x10
Constants.FF_DITAUpdateAllReferences	int <b>Value:</b> (0x01 0x02 0x04 0x08 0x10)
Constants.FP_FirstDITAConrefElementInDoc	int <b>Value:</b> 2861
Constants.FP_FirstDITAXrefElementInDoc	int <b>Value:</b> 2862
Constants.FP_FirstDITALinkElementInDoc	int <b>Value:</b> 2863
Constants.FP_FirstDITATopicrefElementInDoc	int <b>Value:</b> 2864
Constants.FP_FirstDITATopicsetrefElementInDoc	int <b>Value:</b> 2865
Constants.FP_NextDITAConrefElementInDoc	int <b>Value:</b> 2866
Constants.FP_NextDITAXrefElementInDoc	int <b>Value:</b> 2867
Constants.FP_NextDITALinkElementInDoc	int <b>Value:</b> 2868
Constants.FP_NextDITATopicrefElementInDoc	int <b>Value:</b> 2869
Constants.FP_NextDITATopicsetrefElementInDoc	int <b>Value:</b> 2870
Constants.FV_DITA_TYPE_GENERAL	int <b>Value:</b> 1
Constants.FV_DITA_TYPE_RELTABLE	int <b>Value:</b> 2
Constants.FV_DITA_TYPE_KEYDEF	int <b>Value:</b> 3
Constants.FV_DITA_TYPE_TOPICGROUP	int <b>Value:</b> 4
Constants.FV_DITA_TYPE_TOPICHEAD	int <b>Value:</b> 5
Constants.FV_DITA_TYPE_TOPICSETREF	int <b>Value:</b> 6

Property name	Data Type
Constants.FV_DITA_TYPE_TOPICSET	int <b>Value:</b> 7
Constants.FV_DITA_TYPE_MAPREF	int <b>Value:</b> 8
Constants.FV_DITA_TYPE_NAVREF	int <b>Value:</b> 9
Constants.FV_DITA_TYPE_TITLE	int <b>Value:</b> 10
Constants.FV_DITA_TYPE_TOPICREF	int <b>Value:</b> 11
Constants.FV_DITA_TYPE_ANCHOR	int <b>Value:</b> 12
Constants.FV_ENUMERATE_DITACOMP_TOPICREF	int <b>Value:</b> 1 << 1
Constants.FV_ENUMERATE_DITACOMP_MAPREF	int <b>Value:</b> 1 << 2
Constants.FV_ENUMERATE_DITACOMP_TOPICSET	int <b>Value:</b> 1 << 3
Constants.FV_ENUMERATE_DITACOMP_TOPICSETREF	int <b>Value:</b> 1 << 4
Constants.FV_ENUMERATE_DITACOMP_KEYDEF	int <b>Value:</b> 1 << 5
Constants.FV_ENUMERATE_COMP_CHILDMAP_COMPONENTS	int <b>Value:</b> 1 << 6
Constants.FV_ENUMERATE_COMP_CHILDBOOK_COMPONENTS	int <b>Value:</b> 1 << 7
Constants.FV_ENUMERATE_BOOKCOMP_FOLDER	int <b>Value:</b> 1 << 8
Constants.FV_ENUMERATE_BOOKCOMP_GROUP	int <b>Value:</b> 1 << 9
Constants.FV_ENUMERATE_COMP_FOLDER_COMPONENTS	int <b>Value:</b> 1 << 10
Constants.FV_ENUMERATE_COMP_GROUP_COMPONENTS	int <b>Value:</b> 1 << 11
Constants.FV_ENUMERATE_COMPFILE_FMBINARY	int <b>Value:</b> 1 << 12

## Object Reference

Property name	Data Type
Constants.FV_ENUMERATE_COMPFILE_FMMIF	int <b>Value:</b> 1 << 13
Constants.FV_ENUMERATE_COMPFILE_BOOKBINARY	int <b>Value:</b> 1 << 14
Constants.FV_ENUMERATE_COMPFILE_BOOKMIF	int <b>Value:</b> 1 << 15
Constants.FV_ENUMERATE_COMPFILE_XML	int <b>Value:</b> 1 << 16
Constants.FMV_CMDSUPPORTED	int <b>Value:</b> -1
Constants.FF_UNTAGGED_ASK	int <b>Value:</b> 0
Constants.FF_UNTAGGED_UNCOND	int <b>Value:</b> 1
Constants.FF_UNTAGGED_DELETE	int <b>Value:</b> 2
Constants.FS_IteratorItemFieldBookId	int <b>Value:</b> 1
Constants.FS_IteratorItemFieldDocId	int <b>Value:</b> 2
Constants.FS_IteratorItemFieldObjId	int <b>Value:</b> 3
Constants.FS_IteratorItemMaxFields	int <b>Value:</b> 3
Constants.FS_IterParamParseBookInBook	int <b>Value:</b> 1
Constants.FS_IterParamParseMapInBook	int <b>Value:</b> 2
Constants.FS_IterParamSkipExcludedBookComponents	int <b>Value:</b> 3
Constants.FS_IterParamGetBookComponentFolder	int <b>Value:</b> 4
Constants.FS_IterParamGetBookComponentGroup	int <b>Value:</b> 5
Constants.FS_IterParamGetBookComponentRefBook	int <b>Value:</b> 6

<b>Property name</b>	<b>Data Type</b>
Constants.FS_IterParamGetBookComponentRefMap	int <b>Value:</b> 7
Constants.FS_IterParamParseMapInMap	int <b>Value:</b> 8
Constants.FS_IterParamSkipResourceOnlyTopicrefs	int <b>Value:</b> 9
Constants.FS_IterParamSkipNonPrintTopicrefs	int <b>Value:</b> 10
Constants.FS_IterParamSkipTopicsetrefs	int <b>Value:</b> 11
Constants.FS_IterParamGetMapRef	int <b>Value:</b> 12
Constants.FS_NumIteatorParams	int <b>Value:</b> 12
Constants.FV_CMSPopNone	int <b>Value:</b> FV_CMSPopNone
Constants.FV_CMSPopItemUpdated	int <b>Value:</b> FV_CMSPopItemUpdated
Constants.FV_CMSPopDependentsUpdated	int <b>Value:</b> FV_CMSPopDependentsUpdated
Constants.FV_CMSPopDependentsDeleted	int <b>Value:</b> FV_CMSPopDependentsDeleted
Constants.FV_CMSPopItemDeleted	int <b>Value:</b> FV_CMSPopItemDeleted
Constants.FV_CMSPopChildAdded	int <b>Value:</b> FV_CMSPopChildAdded
Constants.FV_CMSPopRootUpdated	int <b>Value:</b> FV_CMSPopRootUpdated
Constants.FP_CMSPopItemProperty_Min	int <b>Value:</b> FP_CMSPopItemProperty_Min
Constants.FP_CMSPopItemProperty_ItemName	int <b>Value:</b> FP_CMSPopItemProperty_ItemName
Constants.FP_CMSPopItemProperty_ItemServerPath	int <b>Value:</b> FP_CMSPopItemProperty_ItemServerPath
Constants.FP_CMSPopItemProperty_ItemLocalPath	int <b>Value:</b> FP_CMSPopItemProperty_ItemLocalPath

<b>Property name</b>	<b>Data Type</b>
Constants.FP_CMSItemProperty_ItemsCheckedOut	int <b>Value:</b> FP_CMSItemProperty_ItemsCheckedOut
Constants.FP_CMSItemProperty_ItemCheckedOutByCurrentUser	int <b>Value:</b> FP_CMSItemProperty_ItemCheckedOutByCurrentUser
Constants.FP_CMSItemProperty_ItemsContainer	int <b>Value:</b> FP_CMSItemProperty_ItemsContainer
Constants.FP_CMSItemProperty_ItemType	int <b>Value:</b> FP_CMSItemProperty_ItemType
Constants.FP_CMSItemProperty_ItemFileType	int <b>Value:</b> FP_CMSItemProperty_ItemFileType
Constants.FP_CMSItemProperty_ItemVersion	int <b>Value:</b> FP_CMSItemProperty_ItemVersion
Constants.FP_CMSItemProperty_Max	int <b>Value:</b> FP_CMSItemProperty_Max
Constants.FV_CMSItemTypeValue_Min	int <b>Value:</b> FV_CMSItemTypeValue_Min
Constants.FV_CMSItemTypeValue_Root	int <b>Value:</b> FV_CMSItemTypeValue_Root
Constants.FV_CMSItemTypeValue_Folder	int <b>Value:</b> FV_CMSItemTypeValue_Folder
Constants.FV_CMSItemTypeValue_File	int <b>Value:</b> FV_CMSItemTypeValue_File
Constants.FV_CMSItemTypeValue_General	int <b>Value:</b> FV_CMSItemTypeValue_General
Constants.FV_CMSItemTypeValue_Max	int <b>Value:</b> FV_CMSItemTypeValue_Max
Constants.FV_CMSItemFileTypeValue_Min	int <b>Value:</b> FV_CMSItemFileTypeValue_Min
Constants.FV_CMSItemFileTypeValue_Xml	int <b>Value:</b> FV_CMSItemFileTypeValue_Xml
Constants.FV_CMSItemFileTypeValue_FmDoc	int <b>Value:</b> FV_CMSItemFileTypeValue_FmDoc
Constants.FV_CMSItemFileTypeValue_Mif	int <b>Value:</b> FV_CMSItemFileTypeValue_Mif
Constants.FV_CMSItemFileTypeValue_DitaMap	int <b>Value:</b> FV_CMSItemFileTypeValue_DitaMap

<b>Property name</b>	<b>Data Type</b>
Constants.FV_CMSItemFileTypeValue_DitaTopic	int <b>Value:</b> FV_CMSItemFileTypeValue_DitaTopic
Constants.FV_CMSItemFileTypeValue_FmBook	int <b>Value:</b> FV_CMSItemFileTypeValue_FmBook
Constants.FV_CMSItemFileTypeValue_Text	int <b>Value:</b> FV_CMSItemFileTypeValue_Text
Constants.FV_CMSItemFileTypeValue_Img	int <b>Value:</b> FV_CMSItemFileTypeValue_Img
Constants.FV_CMSItemFileTypeValue_General	int <b>Value:</b> FV_CMSItemFileTypeValue_General
Constants.FV_CMSItemFileTypeValue_Max	int <b>Value:</b> FV_CMSItemFileTypeValue_Max
Constants.FA_CMSCCommandMin	int <b>Value:</b> FA_CMSCCommandMin
Constants.FA_CMSCreateConnection	int <b>Value:</b> FA_CMSCreateConnection
Constants.FA_CMSSetRootObject	int <b>Value:</b> FA_CMSSetRootObject
Constants.FA_CMSCreateConnMgrUI	int <b>Value:</b> FA_CMSCreateConnMgrUI
Constants.FA_CMSGetItemFromPath	int <b>Value:</b> FA_CMSGetItemFromPath
Constants.FA_CMSCloseConnection	int <b>Value:</b> FA_CMSCloseConnection
Constants.FA_CMSGetCommandMaxOpCode	int <b>Value:</b> FA_CMSGetCommandMaxOpCode
Constants.FA_CMSGetPropertyMaxOpCode	int <b>Value:</b> FA_CMSGetPropertyMaxOpCode
Constants.FA_CMSObjectCheckout	int <b>Value:</b> FA_CMSObjectCheckout
Constants.FA_CMSObjectCheckin	int <b>Value:</b> FA_CMSObjectCheckin
Constants.FA_CMSObjectCancelCheckout	int <b>Value:</b> FA_CMSObjectCancelCheckout
Constants.FA_CMSObjectEdit	int <b>Value:</b> FA_CMSObjectEdit



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

Property name	Data Type
Constants.FA_CMSIsValidCommand	int <b>Value:</b> FA_CMSIsValidCommand
Constants.FA_CMSCommandMax	int <b>Value:</b> FA_CMSCommandMax
Constants.FV_CMSMenu_Min	int <b>Value:</b> FV_CMSMenu_Min
Constants.FV_CMSMenu_Is_Item	int <b>Value:</b> FV_CMSMenu_Is_Item
Constants.FV_CMSMenu_Is_Disabled	int <b>Value:</b> FV_CMSMenu_Is_Disabled
Constants.FV_CMSMenu_Is_Separator	int <b>Value:</b> FV_CMSMenu_Is_Separator
Constants.FV_CMSMenu_Is_SubMenu	int <b>Value:</b> FV_CMSMenu_Is_SubMenu
Constants.FV_CMSMenu_Max	int <b>Value:</b> FV_CMSMenu_Max
Constants.FV_CMSVersion_Min	int <b>Value:</b> FV_CMSVersion_Min
Constants.FV_CMSSameVersion	int <b>Value:</b> FV_CMSSameVersion
Constants.FV_CMSMinorVersion	int <b>Value:</b> FV_CMSMinorVersion
Constants.FV_CMSMajorVersion	int <b>Value:</b> FV_CMSMajorVersion
Constants.FV_CMSVersion_Max	int <b>Value:</b> FV_CMSVersion_Max
Constants.FV_CMSCheckoutUI_Id_ShowDependents	int <b>Value:</b> FV_CMSCheckoutUI_Id_ShowDependents
Constants.FV_CMSCheckinUI_Id_SameVersion	int <b>Value:</b> FV_CMSCheckinUI_Id_SameVersion
Constants.FV_CMSCheckinUI_Id_MinorVersion	int <b>Value:</b> FV_CMSCheckinUI_Id_MinorVersion
Constants.FV_CMSCheckinUI_Id_MajorVersion	int <b>Value:</b> FV_CMSCheckinUI_Id_MajorVersion
Constants.FV_CMSCheckinUI_Id_VersionLabel	int <b>Value:</b> FV_CMSCheckinUI_Id_VersionLabel

Property name	Data Type
Constants.FV_CMSCheckinUI_Id_Description	int <b>Value:</b> FV_CMSCheckinUI_Id_Description
Constants.FV_CMSCheckinUI_Id_CheckinComment	int <b>Value:</b> FV_CMSCheckinUI_Id_CheckinComment
Constants.FV_CMSCheckinUI_Id_MakeThisCurrentVersion	int <b>Value:</b> FV_CMSCheckinUI_Id_MakeThisCurrentVersion
Constants.FV_CMSCDeleteUI_DeleteAllVersion	int <b>Value:</b> FV_CMSCDeleteUI_DeleteAllVersion
Constants.FV_CMSCDeleteUI_DeleteAllDependents	int <b>Value:</b> FV_CMSCDeleteUI_DeleteAllDependents
Constants.FV_CMSCCommandId_Min	int <b>Value:</b> FV_CMSCCommandId_Min
Constants.FV_CMSCCommandNameId	int <b>Value:</b> FV_CMSCCommandNameId
Constants.FV_CMSCCommandConnTypeId	int <b>Value:</b> FV_CMSCCommandConnTypeId
Constants.FV_CMSCCommandServerId	int <b>Value:</b> FV_CMSCCommandServerId
Constants.FV_CMSCCommandUserNameId	int <b>Value:</b> FV_CMSCCommandUserNameId
Constants.FV_CMSCCommandPasswordId	int <b>Value:</b> FV_CMSCCommandPasswordId
Constants.FV_CMSCCommandUserField1	int <b>Value:</b> FV_CMSCCommandUserField1
Constants.FV_CMSCCommandRepositoryId	int <b>Value:</b> FV_CMSCCommandRepositoryId
Constants.FV_CMSCCommandUserField2	int <b>Value:</b> FV_CMSCCommandUserField2
Constants.FV_CMSCCommandFilePathId	int <b>Value:</b> FV_CMSCCommandFilePathId
Constants.FV_CMSCCommandSearchStringId	int <b>Value:</b> FV_CMSCCommandSearchStringId
Constants.FV_CMSCCommandAdvancedSearchStringId	int <b>Value:</b> FV_CMSCCommandAdvancedSearchStringId
Constants.FV_CMSCCommandId	int <b>Value:</b> FV_CMSCCommandId



<b>Property name</b>	<b>Data Type</b>
Constants.FE_CMSSessionFailed	int <b>Value:</b> -115
Constants.FE_CMSSBadCommandId	int <b>Value:</b> -116
Constants.FE_CMSSObjectCreationFailed	int <b>Value:</b> -117
Constants.FE_CMSSRootObjectExists	int <b>Value:</b> -118
Constants.FE_CMSSBadItemFileType	int <b>Value:</b> -119
Constants.FE_CMSSBadItemType	int <b>Value:</b> -120
Constants.FE_CMSSBadItemContainerType	int <b>Value:</b> -121
Constants.FE_CMSSessionCreationFailed	int <b>Value:</b> -122
Constants.FE_CMSSValidCMSCommand	int <b>Value:</b> -123
Constants.FE_CMSSFailedLogin	int <b>Value:</b> -150
Constants.FE_CMSSFailedCheckout	int <b>Value:</b> -151
Constants.FE_CMSSFailedCheckin	int <b>Value:</b> -152
Constants.FE_CMSSFailedCancelCheckout	int <b>Value:</b> -153
Constants.FE_CMSSFailedDelete	int <b>Value:</b> -154
Constants.FE_CMSSFailedOpenFile	int <b>Value:</b> -155
Constants.FE_CMSSFailedUploadObject	int <b>Value:</b> -156

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

## Doc

Property name	Data Type	Description
BannerTextDisplay	Bool	Specifies whether banner text should be displayed in a document. Here is an example of usage:  <code>F_ApiGetInt (FV_SessionId, F_ObjHandleT docId, FP_BannerTextDisplay); F_ApiSetInt (FV_SessionId, F_ObjHandleT docId, FP_BannerTextDisplay, True/False);</code>
CurrentPage	FMOBJECT	Denotes the current page [Body page, Master page or Reference page] in the document.
FirstBodyPageInDoc	BodyPage	Denotes the first body page in the document.
FirstCharFmtInDoc	CharFmt	Indicates the first character tag in the list of the document's character tags.
FirstColorInDoc	Color	Specifies the first color in the list of document's colors.
FirstCombinedFontDefnInDoc	CombinedFontDefn	Specifies the first combined font definition in the list of the document's combined font definitions.
FirstCondFmtInDoc	CondFmt	Indicates the first condition tag in the list of the document's condition tags.
FirstDITAConrefElementInDoc	string	Gets the first Conref element in the document
FirstDITALinkElementInDoc	string	Gets the first Conref element in the document
FirstDITATopicrefElementInDoc	string	Gets the first Topicref element in the document
FirstDITATopicsetrefElementInDoc	string	Gets the first Topicref element in the document
FirstDITAXrefElementInDoc	string	Gets the first cross reference element in the document
FirstFlowInDoc	Flow	Denotes the first flow in the list of the document's flows.
FirstFnInDoc	Fn	Specifies the first footnote in the list of the document's footnotes.
FirstGraphicInDoc	FMOBJECT	Indicates the first graphic object in the list of the document's graphic objects.
FirstMarkerInDoc	Marker	Denotes the first marker in the list of the document's markers.
FirstMarkerTypeInDoc	MarkerType	Specifies the first marker type in the list of the document's marker types.

Property name	Data Type	Description
FirstMasterPageInDoc	<a href="#">MasterPage</a>	Denotes the first master page in the document.
FirstPgFmtInDoc	<a href="#">PgFmt</a>	Indicates the first paragraph tag in the list of the document's paragraph tags.
FirstPgInDoc	<a href="#">Pgf</a>	Specifies the first paragraph in the list of the document's paragraphs.
FirstRefPageInDoc	<a href="#">RefPage</a>	Denotes the first reference page in the document.
FirstRubiInDoc	<a href="#">Rubi</a>	Specifies the first rubi composite in the list of the document's rubi composites.
FirstRulingFmtInDoc	<a href="#">RulingFmt</a>	Indicates the first ruling format in the list of the document's ruling formats.
FirstSelectedTilInDoc	<a href="#">FMObject</a>	Specifies the first selected text inset [(FO_TiApiClient, FO_TiText, FO_TiTextTable, or FO_TiFlow ID)] in the list of selected text insets in the document.
FirstSelectedGraphicInDoc	<a href="#">FMObject</a>	Denotes the first selected graphic object [FO_Graphic ID] in the list of selected graphic objects in the document.
SelectedTbl	<a href="#">Tbl</a>	Denotes the table [FO_Tbl ID] containing the selected table cells.
FirstTblFmtInDoc	<a href="#">TblFmt</a>	Denotes the first table format [FO_TblFmt ID] in the list of the document's table formats.
FirstTblInDoc	<a href="#">Tbl</a>	Denotes the first table [FO_Tbl ID] in the list of the document's tables.
FirstTilInDoc	<a href="#">FMObject</a>	Denotes the first text inset [FO_TiApiClient, FO_TiText, FO_TiTextTable, or FO_TiFlow ID] in the list of the document's text insets.
FirstVarFmtInDoc	<a href="#">VarFmt</a>	Specifies the first variable format [FO_VarFmt ID] in the list of the document's variable formats.
FirstVarInDoc	<a href="#">Var</a>	Indicates the first variable [FO_Var ID] in the list of the document's variables.
FirstXRefFmtInDoc	<a href="#">XRefFmt</a>	Denotes the first cross-reference format [FO_XRefFmtID] in the list of the document's cross reference formats.
FirstXRefInDoc	<a href="#">XRef</a>	Specifies the first cross-reference [FO_XRef ID] in the list of the document's cross-references.
HiddenPage	<a href="#">HiddenPage</a>	Denotes a hidden page [FO_HiddenPage ID] in the document.
KeyCatalog	<a href="#">BodyPage</a>	Indicates the last body page [FO_BodyPage ID] in the document.
LastBodyPageInDoc	<a href="#">BodyPage</a>	Indicates the last body page [FO_BodyPage ID] in the document.
LastMasterPageInDoc	<a href="#">MasterPage</a>	Indicates the last master page [FO_MasterPage ID] in the document.
LastRefPageInDoc	<a href="#">RefPage</a>	Specifies the last reference page [FO_RefPage ID] in the document.
LeftMasterPage	<a href="#">MasterPage</a>	Denotes the left master page [FO_MasterPage ID] in the document.

Property name	Data Type	Description
MainFlowInDoc	Flow	Specifies the main flow [FO_Flow ID] in the document.
MarkerTypeNames	Strings	Indicates the list of marker type names.
NextOpenDocInSession	Doc	Denotes the next open document [(FO_Doc ID) in the list of open documents in the session.
RightMasterPage	MasterPage	Denotes the right master page [FO_MasterPage ID] in the document.
AcrobatBookmarkDisplayTags	int	Set to 1 if Include Paragraph Tags in Bookmark Text is on (the paragraph tag is added before the paragraph text in each bookmark); 0, otherwise.
DocAcrobatColumnArticleThreads	int	Set to 1 if you want separate article threads for each column; 0 if you want separate article threads for each text frame.  The DocPDFNoArticleThread property must be set to False, if the DocAcrobatColumnArticleThreads property is set to True.
DocAcrobatDefaultsChanged	int	Set to 1 if the default heuristics for determining the paragraph level are disabled; 0 otherwise.
DocAcrobatElementList	Strings	Denotes the list of the element tags and context labels to include in bookmarks. The DocPDFElementList property applies only to structured FrameMaker documents.
DocAcrobatElements	int	Set to 1 if elements rather than paragraphs are used for bookmarks; set to 0 otherwise. The DocPDFElements property applies only to structured FrameMaker documents.
DocAcrobatNoArticleThreads	int	Set to 1 if you do not want article threads in the resulting PDF; set to 0 otherwise.
GenerateAcrobatInfo	int	Set to 1 if Generate Adobe Acrobat Data is on. To generate PDF data, you must set the other document print properties as follows: <ul style="list-style-type: none"> <li>• PrintToFile - True</li> <li>• PrintThumbnails - False</li> <li>• PrintSeps - False</li> <li>• PrintBlankPages - True</li> <li>• PrintLastSheetFirst - False</li> <li>• PrintNumCopies - 1</li> <li>• PrintOddPages - True</li> <li>• PrintEvenPages - True</li> <li>• PrintScale - 100%</li> </ul>
PDFAllNamedDestinations	int	Set to 1 to create named destinations for all paragraphs and elements in the PDF file generated from this document.
PDFBookmark	bool	Set to True if FrameMaker is to generate bookmarks when saving as PDF; set to False otherwise.



Property name	Data Type	Description
PDFBookmarksOpenLevel	int	Denote the level of bookmarks to have expanded when Acrobat opens the generated PDF document. The value can be any integer, or one of the following defined values: <ul style="list-style-type: none"> <li>• Constants.FV_PDFBookmarksOpenDefaultLevel (-1)</li> <li>• Constants.FV_PDFBookmarksOpenAllLevels (-2)</li> <li>• Constant.FV_PDFBookmarksOpenNoneLevel (-3)</li> </ul> If you specify an integer greater than the number of levels in the Bookmarks Settings, Constants.FV_PDFBookmarksOpenAllLevels (-1) takes effect.
PDFConvertCMYKtoRGB	int	When 1, the PDFConvertCMYKtoRGB setting corresponds with setting Convert CMYK colors to RGB in the Save As PDF dialog box.
PDFDestsMarked	int	Set to 1 if the document has paragraphs or elements marked via the MarkedForNamed Destination property.  One of two things must happen in order for the PDFDestsMarked property to be True: <ol style="list-style-type: none"> <li>1 The document must be created in version 6.0 or later; the document must be opened in version 6.0 or later</li> <li>2 The PDF FileSize Optimization client must be run over it to mark all paragraphs or elements that are targets of hypertext links.</li> </ol> Normally, your client should not set this value.
PDFDistillerAbsent	int	A value of 1 indicates that Acrobat Distiller is not available. Any other value indicates that Acrobat Distiller is available.
PDFDocInfo	Strings	Contains a list of strings expressing values to be set in the PDF Document Info dictionary when you save the book as PDF.  Each dictionary entry is expressed as a pair of strings; the first string expresses the field name, and the second string expresses the field value.
PDFEndPage	string	Indicates then last page of the printing page range, in the FrameMaker numbering style.
PDFJobOption	string	Denotes the name of the Distiller Job Option. If the specified name does not exist in the Distiller Job Options list, then the first Distiller Job Option in the list is used.
PDFJobOptionsAbsent	int	A value of 1 indicates that PDF Job Options are not available.
PDFOpenPage	string	Denote the PDF page number, in the FrameMaker numbering style, at which Acrobat opens the generated PDF document.
PDFPageHeight	int	Indicates the page height for the generated PDF.
PDFPageWidth	int	Indicates the page width for the generated PDF.

Property name	Data Type	Description
PDFPrintPageRange	int	Set to 1 to generate PDF for the specified page range; if set to 0, FrameMaker generates PDF for the entire document or book.
PDFRegistrationMarks	int	Specifies the registration marks for the generated PDF. May be one of: <ul style="list-style-type: none"> <li>• Constants.FV_PDFRegistrationMarksNone(0)</li> <li>• Constants.FV_PDFRegistrationMarksWestern (1)</li> <li>• Constants.FV_PDFRegistrationMarksTombo (2)</li> </ul>
PDFSeparateFiles	int	Set to 1, if a separate PDF file should be generated for each document in a book. This property can be set for single document, but is ignored in that case.
PDFStartPage	string	Denotes the first page of the printing page range, in the FrameMaker numbering style.
PDFStructure	int	Set to 1 to generate tagged or structured PDF.
PDFZoomFactor	int	When the PDFZoomType property is set to Constants.FV_PDFZoomNone (0), the PDFZoomFactor property denotes the zoom percentage of the PDF document (metric 25% to 1600%).  If the value of this property is negative or zero, Constants.FV_PDFZoomDefault (1) takes effect.
PDFZoomType	int	Indicates the PDF zoom setting with which Acrobat opens the generated PDF document. The value can be one of: <ul style="list-style-type: none"> <li>• Constants.FV_PDFZoomDefault (1)</li> <li>• Constants.FV_PDFZoomPage (2)</li> <li>• Constants.FV_PDFZoomWidth (3)</li> <li>• Constants.FV_PDFZoomHeight (4)</li> <li>• Constants.FV_PDFZoomNone (0)</li> </ul> If a different value is specified, Constants.FV_PDFZoomDefault (1) takes effect.
ChapNumComputeMethod	int	Denotes the component document's chapter numbering type. The possible value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_NUM_CONTINUE (0x01): Continue the numbering from the previous chapter.</li> <li>• Constants.FV_NUM_RESTART (0x02): Use the value specified for the ChapterNumber property.</li> <li>• Constants.FV_NUM_SAME (0x03): Use the same chapter number as for the previous file.</li> <li>• Constants.FV_NUM_READ_FROM_FILE (0x00): Use the numbering properties from the document associated with this book component.</li> </ul>
ChapterNumber	int	Use the value of the ChapterNumber property as the chapter number, if the ChapNumComputeMethod property is set to Constants.FV_NUM_RESTART (0x02).

Property name	Data Type	Description
ChapterNumStyle	int	The numbering style. The possible value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_NUMSTYLE_NUMERIC (0x00): Arabic.</li> <li>• Constants.FV_NUMSTYLE_ROMAN_UC (0x01): Roman, uppercase.</li> <li>• Constants.FV_NUMSTYLE_ROMAN_LC (0x02): Roman, lowercase.</li> <li>• Constants.FV_NUMSTYLE_ALPHA_UC (0x03): Alphabetic, uppercase.</li> <li>• Constants.FV_NUMSTYLE_ALPHA_LC (0x04): Alphabetic, lowercase.</li> <li>• Constants.FV_NUMSTYLE_KANJI (0x05): Kanji.</li> <li>• Constants.FV_NUMSTYLE_ZENKAKU (0x06): Zenkaku.</li> <li>• Constants.FV_NUMSTYLE_ZENKAKU_UC (0x07): Zenkaku, uppercase.</li> <li>• Constants.FV_NUMSTYLE_ZENKAKU_LC (0x08): Zenkaku, lowercase.</li> <li>• Constants.FV_NUMSTYLE_KANJI_KAZU (0x09): Kazu.</li> <li>• Constants.FV_NUMSTYLE_DAIJI (0x0a): Daiji.</li> <li>• Constants.FV_NUMSTYLE_TEXT (0x0b): Text.</li> </ul>
ChapterNumText	string	If the ChapNumStyle property is set to Constants.FV_NUMSTYLE_TEXT (0x0b), use the value of the ChapterNumText property as the chapter number.
Dictionary	Strings	List of words to accept when spell-checking the document.
DocIsModified	int	Set to 1 if the document has been modified. While this property is read-only, you can modify a document without setting this property to True by setting the Untouchable property to True for the document before your client modifies it. Otherwise, set to 0.
DocIsViewOnly	int	Set to 1 if the document is View Only, else set to 0.
DocOpenType	int	Denotes the type of document opened. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_DOC_TYPE_BINARY (0x01): Frame binary document</li> <li>• Constants.FV_DOC_TYPE_TEXT (0x02): ASCII text document</li> <li>• Constants.FV_DOC_TYPE_MIF (0x03): MIF document</li> <li>• Constants.FV_DOC_TYPE_FILTER (0x04): A filtered document.</li> </ul>

Property name	Data Type	Description
DocSaveType	int	Denotes the type of document saved. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_DOC_TYPE_BINARY (0x01): Frame binary document</li> <li>• Constants.FV_DOC_TYPE_TEXT (0x02): ASCII text document</li> <li>• Constants.FV_DOC_TYPE_MIF (0x03): MIF document</li> <li>• Constants.FV_DOC_TYPE_FILTER (0x04) : A filtered document.</li> </ul>
DontUpdateTextInsets	int	Set to 1 if FrameMaker is not to automatically update text insets when it opens the document.
DontUpdateXRefs	int	Set to 1 if FrameMaker is not to automatically update cross-references when it opens or prints the document.
FormatOverride	int	Specifies whether there are format overrides at the current insertion point.  If the insertion point is in a text range that has a character format applied to it, the FormatOverride property is True if (and only if) the text formatting at the insertion point overrides the character format.  If the insertion point is in a text range that does not have a character format applied to it, the FormatOverride property is True if (and only if) the paragraph containing the insertion point has formatting that overrides the Paragraph Catalog format.
IsOnScreen	int	Returns 1 if the document is visible on the screen, otherwise returns 0.
LineNumDistance	Metric	Sets the line number display width, that is, the space in which the line numbers are displayed.
LineNumRestart	int	If set, restarts line number display on each page.
LineNumShow	int	If set, enables the line number display.
Name	string	Denotes the filename of the document.
PageNumComputeMethod	int	Denotes the component document's page numbering type. The possible value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_NUM_CONTINUE (0x01): Continue the numbering from the previous file.</li> <li>• Constants.FV_NUM_RESTART (0x02): Restart numbering at the value specified by the FirstPageNum property.</li> </ul>
PgfNumComputeMethod	int	Denotes the document's paragraph numbering type. The possible value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_NUM_CONTINUE (0x01): Continue the numbering from the previous file.</li> <li>• Constants.FV_NUM_RESTART (0x02): Restart the numbering at 1.</li> </ul>
SpecifiedKeyCatalog	F_ObjHandleT	Key Catalog specified for using for the document.

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

Property name	Data Type	Description
VolumeNumStyle	int	Denotes the volume numbering style. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_NUMSTYLE_NUMERIC (0x00): Arabic</li> <li>Constants.FV_NUMSTYLE_ROMAN_UC (0x01): Roman numerals, uppercase.</li> <li>Constants.FV_NUMSTYLE_ROMAN_LC (0x02): Roman numerals, lowercase.</li> <li>Constants.FV_NUMSTYLE_ALPHA_UC (0x03): Alphabetic, uppercase.</li> <li>Constants.FV_NUMSTYLE_ALPHA_LC (0x04): Alphabetic, lowercase.</li> <li>Constants.FV_NUMSTYLE_KANJI (0x05): Kanji</li> <li>Constants.FV_NUMSTYLE_ZENKAKU (0x06): Zenkaku</li> <li>Constants.FV_NUMSTYLE_ZENKAKU_UC (0x07): Zenkaku, uppercase.</li> <li>Constants.FV_NUMSTYLE_ZENKAKU_LC (0x08): Zenkaku, lowercase.</li> <li>Constants.FV_NUMSTYLE_KANJI_KAZU (0x09): Kazu</li> <li>Constants.FV_NUMSTYLE_DAIJI (0x0a): Daiji</li> <li>Constants.FV_NUMSTYLE_TEXT (0x0b): Text</li> </ul>
VolumeNumText	string	Use the value of the VolumeNumText property, if the VolNumStyle property is set to Constants.FV_NUMSTYLE_TEXT (0x0b).
AutoChangeBars	int	Returns 1 if Automatic Change Bars are enabled in the document.
ChangeBarColor	Color	Indicates the spot color (FO_Color ID).
ChangeBarDistance	int	Denotes the distance between change bar and text column.
ChangeBarPosition	int	Denotes the position of the change bars. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CB_COL_LEFT (0x00): Left of Column</li> <li>Constants.FV_CB_COL_RIGHT (0x01): Right of Column</li> <li>Constants.FV_CB_COL_NEAREST (0x02): Side nearest to Page Edge</li> <li>Constants.FV_CB_COL_FURTHEST (0x03): Side farthest from Page Edge</li> </ul>
ChangeBarThickness	int	Denotes the width of the change bars.
ShowAll	int	Set to 1 to display all document conditions, else set to 0.
ShowCondIndicators	int	Set to 1 to display all condition indicators (Format Overrides). Else, set to 0.
EqnIntegralSizeLarge	int	Indicates the point size of integral symbol in large equations (2 pt to 400 pt).

Property name	Data Type	Description
EqnIntegralSizeMed	int	Indicates the point size of integral symbol in medium equations (2 pt to 400 pt).
EqnIntegralSizeSmall	int	Indicates the point size of integral symbol in small equations (2 pt to 400 pt).
EqnLevel1SizeLarge	int	Indicates the point size of level 1 expressions in large equations (2 pt to 400 pt).
EqnLevel1SizeMed	int	Indicates the point size of level 1 expressions in medium equations (2 pt to 400 pt).
EqnLevel1SizeSmall	int	Indicates the point size of level 1 expressions in small equations (2 pt to 400 pt).
EqnLevel2SizeLarge	int	Indicates the point size of level 2 expressions in large equations (2 pt to 400 pt).
EqnLevel2SizeMed	int	Indicates the point size of level 2 expressions in medium equations (2 pt to 400 pt).
EqnLevel2SizeSmall	int	Indicates the point size of level 2 expressions in small equations (2 pt to 400 pt).
EqnLevel3SizeLarge	int	Indicates the point size of level 3 expressions in large equations (2 pt to 400 pt).
EqnLevel3SizeMed	int	Indicates the point size of level 3 expressions in medium equations (2 pt to 400 pt).
EqnLevel3SizeSmall	int	Indicates the point size of level 3 expressions in small equations (2 pt to 400 pt).
EqnSigmaSizeLarge	int	Indicates the point size of the sigma symbol in large equations (2 pt to 400 pt).
EqnSigmaSizeMed	int	Indicates the point size of the sigma symbol in medium equations (2 pt to 400 pt).
EqnSigmaSizeSmall	int	Indicates the point size of the sigma symbol in small equations (2 pt to 400 pt).
Functions	string	Denotes the character format tag of the equation font to apply to Math Functions.
Numbers	string	Denotes the character format tag of the equation font to apply to Math Numbers.
Strings	string	Denotes the character format tag of the equation font to apply to Math Strings.
Symbols	string	Denotes the character format tag of the equation font to apply to Math Symbols.
SymbolsList	Strings	List of math symbol fonts used in the Equation Fonts dialog box.
Variables	string	List of math symbol fonts used in the Equation Fonts dialog box.
HypertextDoValidate	bool	Set to 1 if the next hypertext string sent to the HypertextCommandText property is to be validated.
HypertextCommandText	string	Denotes the hypertext command to parse. Setting this value executes the parser. If the HypertextDoValidate property is True, the command is parsed and validated.

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



Property name	Data Type	Description
HypertextValidateErr	int	<p>Set to a non-zero value if the <code>HypertextDoValidate</code> property is set to <code>True</code> and there is a validation error.</p> <p>The <code>HypertextValidateErr</code> property is set to one of the following values if there is a validation error:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_HypertextValid (0)</code> - No validation errors</li> <li>• <code>Constants.FV_HypertextUsesDefaultText (200)</code> - Default text was found as an argument; are you sure the default text is what you want?</li> <li>• <code>Constants.FV_HypertextFileNotRegular (210)</code> - The referenced file could not be found, or is not a regular file; for example, it could be a directory name.</li> <li>• <code>Constants.FV_HypertextFileNotMakerDoc (211)</code> - The referenced file is not created by FrameMaker.</li> <li>• <code>Constants.FV_HypertextCantOpenDestFile (212)</code> - Cannot open the file; perhaps you do not have permission, or the file is locked.</li> <li>• <code>Constants.FV_HypertextDestinationLinkNotFound (220)</code> - The referenced file is valid, but cannot find the named link within it.</li> <li>• <code>Constants.FV_HypertextPageNameNotFound (230)</code> - The referenced file is valid, but the specified page is not found.</li> <li>• <code>Constants.FV_HypertextUnrecognizedObjectType (240)</code> - The referenced file is valid, but the link is to an object with an unrecognized object type.</li> <li>• <code>Constants.FV_HypertextObjectIDNotFound (241)</code> - The file is a link to an object, but cannot find the linked object.</li> <li>• <code>Constants.FV_HypertextBadMatrixSize (250)</code> - One or both of the matrix dimensions are illegal; The values must be between 1 and 99.</li> <li>• <code>Constants.FV_HypertextMatrixCommandInvalid (251)</code> - One of the commands in the reference page flow for a matrix command has a parse or validation error.</li> <li>• <code>Constants.FV_HypertextFlowMissingLines (252)</code> - The reference flow for a matrix or popup command is missing one or more lines.</li> <li>• <code>Constants.FV_HypertextNoNamedFlow (260)</code> - Cannot find the named reference flow for a matrix or popup command.</li> <li>• <code>Constants.FV_HypertextRecursiveFlow (261)</code> - The reference flow for a matrix or popup command contains nested popup or matrix commands that name a parent reference flow.</li> <li>• <code>Constants.FV_HypertextMissingPopupMarker (270)</code> - At least one entry in the popup command's reference flow has no hypertext marker in it.</li> <li>• <code>Constants.FV_HypertextMissingPopupLabelItem (271)</code> - One entry in the popup command's reference flow has no text in it.</li> <li>• <code>Constants.FV_HypertextEmptyLineInMiddleOfPopup (272)</code> - One entry in the popup command's reference flow has no text in it.</li> </ul>

Property name	Data Type	Description
HypertextParseBadParam	int	Denotes the index into the string list contained in the HypertextParsedArgs property, in case of a parser error.
HypertextParseErrMsg	string	Denotes the message that FrameMaker generates for a parser error.

Property name	Data Type	Description
HypertextParsedCmdCode	int	<p>Denotes the FrameMaker hypertext command in the HypertextCommandText property, as determined by the parser.</p> <p>The possible value for the HypertextParsedCmdCode property is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_CmdError (1) -Parser is in an error state</li> <li>• Constants.FV_CmdUnknown (2) - Unknown command</li> <li>• Constants.FV_CmdNoop (3) - Command causes no event.</li> <li>• Constants.FV_CmdAlert (8) - alert command</li> <li>• Constants.FV_CmdAlertTitle (9) - alerttitle command</li> <li>• Constants.FV_CmdExit (10) - exit command</li> <li>• Constants.FV_CmdGoToLink (11) - gotolink command</li> <li>• Constants.FV_CmdGoToLinkFitWin (12) - gotolinkfitwin command</li> <li>• Constants.FV_CmdGoToNew (13) - gotonew command</li> <li>• Constants.FV_CmdGoToPage (14) - gotopage command</li> <li>• Constants.FV_CmdGoToObjectId (15) - gotoObjectId command</li> <li>• Constants.FV_CmdGoToObjectIdFitWin (16) - gotoObjectIdfitwin command</li> <li>• Constants.FV_CmdMatrix (17) - matrix command</li> <li>• Constants.FV_CmdMessage (18) - message command</li> <li>• Constants.FV_CmdNewLink (19) - newlink command</li> <li>• Constants.FV_CmdNextPage (20) - nextpage command</li> <li>• Constants.FV_CmdPreviousPage (21) - previouspage command</li> <li>• Constants.FV_CmdOpenLink (22) - openlink command</li> <li>• Constants.FV_CmdOpenLinkFitWin (23) - openlinkfitwin command</li> <li>• Constants.FV_CmdOpenNew (24) - opennew command</li> <li>• Constants.FV_CmdOpenObjectId (25) - openObjectId command</li> <li>• Constants.FV_CmdOpenObjectIdFitWin (26) - openObjectIdfitwin command</li> <li>• Constants.FV_CmdOpenPage (27) - openpage command</li> <li>• Constants.FV_CmdPopup (28) - popup command</li> <li>• Constants.FV_CmdPreviousLink (29) - previouslink command</li> </ul>

Property name	Data Type	Description
HypertextParsedCmdDest	int	<p>Denotes the destination type for link commands in the HypertextCommandText property, as determined by the parser.</p> <p>The possible value for the HypertextParsedCmdDest property is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_DestNowhere (0) - No destination found</li> <li>• Constants.FV_DestMarkerNewLink (1) - Destination is a newlink</li> <li>• Constants.FV_DestFirstPage (2) - Destination is the first page of a file</li> <li>• Constants.FV_DestLastPage (3) - Destination is the last page of a file</li> <li>• Constants.FV_DestPageNum (4) - Destination is a named page (usually a page number)</li> <li>• Constants.FV_DestFluidFlow (5) - Destination is to a fluid flow document</li> <li>• Constants.FV_DestMarker (6) - Destination is a marker</li> <li>• Constants.FV_DestObjectId (7) - Destination is an object ID (usually for generated hypertext commands)</li> <li>• Constants.FV_DestXRef (8) - Destination is a cross-reference</li> </ul>
HypertextParsedCmdDestObjType	int	<p>Denotes the type of the object in the target document, for links to objects.</p> <p>The possible value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_ObjectUnknown (0) - Unknown or invalid object</li> <li>• Constants.FV_ObjectMarker (1) - Object is a marker</li> <li>• Constants.FV_ObjectPgf (2) - Object is a paragraph</li> <li>• Constants.FV_ObjectXref (3) - Object is a cross-reference</li> <li>• Constants.FV_ObjectGraphic (4) - Object is a graphic</li> <li>• Constants.FV_ObjectElement (5) - Object is an element</li> <li>• Constants.FV_ObjectTextInset (6) - Object is a text inset</li> <li>• Constants.FV_ObjectDataLink (7) - Object is subscribed data</li> </ul>
HypertextParsedCmdDestObjID	int	Denotes the object in the target document for links to objects.
HypertextParsedCmdMatrixRows	int	Denotes the number of rows in the matrix, if the value of the HypertextParsedCmdCode property is set to Constants.FV_CmdMatrix (17)
HypertextParsedCmdMatrixColumns	int	Denotes the number of columns in the matrix, if the value of the HypertextParsedCmdCode property is set to Constants.FV_CmdMatrix (17)

Property name	Data Type	Description
HypertextParsedLinkName	string	Indicates either the value of a newlink command, or a keyword such as FirstPage or LastPage, for links to new targets.
HypertextParsedPageName	string	Denotes the pagenumber, for links to pages.
HypertextParsedFlowName	string	Denotes the name of the flow (on a reference page) that contains the popup or matrix list of commands, for popup and matrix commands.
HypertextParsedClientName	string	Indicates the name of the API client to receive the message, for message commands.
HypertextParsedTitle	string	Indicates the specified title for the alert box, if the value of the HypertextParsedCmdCode property is set to Constants.FV_CmdAlertTitle (9)
HypertextParsedMessage	string	Denotes the specified message for the hypertext command, if the value of the HypertextParsedCmdCode command is set to Constants.FV_CmdAlert (8), Constants.FV_CmdAlertTitle (9) or Constants.FV_CmdMessage (18).
HypertextParsedDIFilename	string	Denotes the absolute path to the target file, expressed in platform independent syntax, for links to external files.
MenuBar	Menu	Denotes the identifier of the document's menu bar (FO_Menu ID).
ViewOnlyMenuBar	Menu	Denotes the identifier of the document's menu bar when the document is locked (FO_Menu ID).
FnCustNumString	string	Specifies the characters for custom document footnote numbers.
FnFirstNum	int	Indicates the first document footnote number.
FnFmt	string	Indicates the footnote paragraph tag.
FnHeightPerCol	int	Denotes the maximum height allowed for document footnotes. The permissible range of values is 36 points to 32767 points.
FnInstancePosition	int	Specifies the placement of the footnote number in the footnote. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_FN_POS_SUPER (0x00) - Superscript</li> <li>• Constants.FV_FN_POS_BASELINE (0x01) - Baseline</li> <li>• Constants.FV_FN_POS_SUB (0x02) - Subscript</li> </ul>
FnInstancePrefix	string	Indicates the prefix to appear before the document footnote number in the footnote.
FnInstanceSuffix	string	Indicates the suffix to appear after the document footnote number in the footnote.

Property name	Data Type	Description
FnNumComputeMethod	int	The document's footnote numbering type. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_NUM_CONTINUE (0x01) - Continue the numbering from the previous file.</li> <li>• Constants.FV_NUM_RESTART (0x02) - Restart numbering at the value specified by the associated FO_Doc object's FnFirstNum property.</li> <li>• Constants.FV_NUM_PERPAGE (0x03) - Restart numbering on each page.</li> </ul>
FnNumStyle	int	Indicates the document footnote numbering style. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_FN_NUM_NUMERIC (0x00) - Arabic</li> <li>• Constants.FV_FN_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>• Constants.FV_FN_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>• Constants.FV_FN_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_FN_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>• Constants.FV_FN_NUM_KANJI (0x06) - Kanji characters</li> <li>• Constants.FV_FN_NUM_ZENKAKU (0x07) - Zenkaku</li> <li>• Constants.FV_FN_NUM_ZENKAKU_UC (0x08) - Zenkaku uppercase</li> <li>• Constants.FV_FN_NUM_ZENKAKU_LC (0x09) - Zenkaku lowercase</li> <li>• Constants.FV_FN_NUM_KANJI_KAZU (0x0A) - Kazu</li> <li>• Constants.FV_FN_NUM_DAIJI (0x0B) - Daiji</li> <li>• Constants.FV_FN_NUM_CUSTOM (0x05) - Custom numbering</li> </ul>
FnRefPosition	int	Specifies the placement of the footnote reference in the footnote. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_FN_POS_SUPER (0x00) - Superscript</li> <li>• Constants.FV_FN_POS_BASELINE (0x01) - Baseline</li> <li>• Constants.FV_FN_POS_SUB (0x02) - Subscript</li> </ul>
FnRefPrefix	string	Indicates the prefix to appear before the number in the document text.
FnRefSuffix	string	Indicates the suffix to appear after the number in the document text.
BottomMargin	int	Indicates the bottom page margin.
ColGap	int	Denotes the size of the gap between text columns

Property name	Data Type	Description
DocIsDoubleSided	int	Set to 1 if the layout of the document is a 2 sided page; 0 otherwise.
FirstPageNum	int	Indicates the page number of the first page.
FirstPageVerso	int	Set to 1 for right first page; 0 for left first page.
LeftMargin	int	Denotes the left page margin.
NumCols	int	Indicates the number of columns on the page.
PageHeight	int	Specifies the height of the document's pages.  <i>Note: Setting this property, automatically sets the page height for all the body pages of the document.</i>
PageNumStyle	int	Denotes the page numbering style. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PAGE_NUM_NUMERIC (0x01) - Arabic</li> <li>• Constants.FV_PAGE_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>• Constants.FV_PAGE_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>• Constants.FV_PAGE_NUM_KANJI (0x05) - Kanji characters</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU (0x06) - Zenkaku</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU_UC (0x07) - Zenkaku uppercase</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU_LC (0x08) - Zenkaku lowercase</li> <li>• Constants.FV_PAGE_NUM_KANJI_KAZU (0x09) - Kazu</li> <li>• Constants.FV_PAGE_NUM_DAIJI (0x0a) - Daiji</li> </ul>
PageRounding	int	Indicates how to round pages. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PR_DEL_EMPTY (0x01) - Delete Empty Pages</li> <li>• Constants.FV_PR_KEEP_NUM_EVEN (0x02) - Make Page Count Even</li> <li>• Constants.FV_PR_KEEP_NUM_ODD (0x03) - Make Page Count Odd</li> <li>• FV_PR_DONT_CHANGE (0x04) - Do Not Change Page Count</li> </ul>

Property name	Data Type	Description
PageWidth	int	Specifies the width of the document's pages.  <i>Note: Setting this property, automatically sets the page width for all the body pages of the document.</i>
PointPageNumStyle	int	Denotes the point page numbering style. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PAGE_NUM_NUMERIC (0x01) - Arabic</li> <li>• Constants.FV_PAGE_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>• Constants.FV_PAGE_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_PAGE_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>• Constants.FV_PAGE_NUM_KANJI (0x05) - Kanji characters</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU (0x06) - Zenkaku</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU_UC (0x07) - Zenkaku uppercase</li> <li>• Constants.FV_PAGE_NUM_ZENKAKU_LC (0x08) - Zenkaku lowercase</li> <li>• Constants.FV_PAGE_NUM_KANJI_KAZU (0x09) - Kazu</li> <li>• Constants.FV_PAGE_NUM_DAIJI (0x0a) - Daiji</li> </ul>
RightMargin	int	Denotes the Right page margin.
SmartQuotes	int	Returns 1 if Smart Quotes is enabled; returns 0 otherwise.
SmartSpaces	int	Returns 1 if Smart Spaces is enabled; returns 0 otherwise.
TopMargin	int	Denotes the Top Page Margin.
PrintBlankPages	int	Returns 1 if the PageRounding property allows an empty page at end of a document. Returns 0, otherwise.
PrintCollated	int	Returns 1 if Collate is enabled. Returns 0, otherwise.
PrintCols	int	Indicates the number of columns to print, if the PrintThumbnails property is set to True.
PrintEmulsion	int	Specifies the direction of the print emulsion. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_EMUL_UP (0) - Emulsion side up</li> <li>• Constants.FV_EMUL_DOWN (1) - Emulsion side down</li> </ul>



Property name	Data Type	Description
PrintEndPage	int	Specifies the number of the last page to print.  <i>Note: The value of the DocFluidFlow property must be 0, when you use the PrintEndPage property. You cannot print a range of pages when a document is in fluid view.</i>
PrintEndPageName	int	Denotes the page number string for the last page to be printed. Use this property when the pages are numbered with a style other than <code>Constants.FV_PAGE_NUM_NUMERIC (0x01)</code> .  <i>Note: The value of the DocFluidFlow property must be 0, when you use the PrintEndPageName property. You cannot print a range of pages when a document is in fluid view.</i>
PrintEndPoint	int	Denotes the number of the last point page to print.
PrinterName	string	Specifies the name of the printer.  <i>Note: Setting the PrinterName property on Windows has no effect.</i>  <i>Note: You can set the printer to the default printer by specifying the PrinterName as null.</i>
PrintEvenPages	int	Set to 1 if Print Even-Numbered Pages is enabled. Set to 0, otherwise.
PrintFileName	string	Name of the file to which to print. You can set the filename to the default filename by specifying the value of the PrintFileName property as null.
PrintImaging	int	Specifies the type of print imaging. The value is one of: <ul style="list-style-type: none"><li>• <code>Constants.FV_IMG_POSITIVE (0)</code></li><li>• <code>Constants.FV_IMG_NEGATIVE (1)</code></li></ul>
PrintLastSheetFirst	int	Set to 1 to print the last sheet first. Else, set to 0.
PrintLowRes	int	Set to 1 to print in low resolution. Else, set to 0.
PrintManualFeed	int	Set to 1 if Manual feed is enabled. Else, set to 0.
PrintNumCopies	int	Denotes the number of copies to print.
PrintOddPages	int	Set to 1 if Print Odd-Numbered Pages is enabled. Else, set to 0.
PrintPaperHeight	int	Specifies the height of the printing paper.
PrintPaperWidth	int	Indicates the width of the printing paper.
PrintRegistrationMarks	int	Set to 1 if Registration Marks is enabled; Set to 0 otherwise.
PrintRows	int	Indicates the number of rows to print if the PrintThumbnails property is set to True.

Property name	Data Type	Description
PrintScale	int	Denotes the Print scale factor expressed as a percentage metric (0% to 100%).For print scale percentages, the value <<16 or 0x10000 specifies 100%.
PrintScope	int	Denotes the pages to print. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PR_ALL (1) - Print all pages</li> <li>• Constants.FV_PR_RANGE (2) - Print a range of pages</li> </ul> <p><b>Note:</b> The value of the DocFluidFlow property must be set to 0 when you set the PrintScope property. You cannot print a range of pages when a document is in the fluid view.</p>
PrintSeps	int	Returns 1 if Print Separations is enabled. Else, returns 0.
PrintStartPage	int	Denotes the number of the first page to print. <p><b>Note:</b> The value of the DocFluidFlow property must be set to 0 when you set the PrintStartPage property. You cannot print a range of pages when a document is in the fluid view.</p>
PrintStartPageName	int	Indicates the page number string for the first page to print.This property is valid only when the pages are numbered with a style other than Constants.FV_PAGE_NUM_NUMERIC (0x00) <p><b>Note:</b> The value of the DocFluidFlow property must be set to 0 when you set the PrintStartPage property. You cannot print a range of pages when a document is in the fluid view.</p>
PrintStartPoint	int	Indicates the number of the first point page to print.
PrintThumbnails	int	Returns 1 if Print Thumbnails is enabled. Returns 0, otherwise.
PrintToFile	int	Returns 1 if Print Only to File is enabled. Else, returns 0.
SkipBlankSeps	int	Returns 1 if Skip Blank Separations (do not print blank color separations) is enabled. Else, returns 0.
TrapwiseCompatibility	bool	Set to 1 if Trapwise Compatibility is enabled. Else, set to 0. <p><b>Note:</b> Setting the TrapwiseCompatibility parameter to True automatically sets the PrintToFile parameter to True and the PrintSeps parameter to False.</p>
NarrowRubiSpaceForOther	int	Denotes the value for a composite ruby object. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_Wide (0)</li> <li>• Constants.FV_Narrow (1)</li> <li>• Constants.FV_Proportional (2)</li> </ul>
RubiOverhang	int	Set to 1 if Rubi is allowed to overhang, else set to 0.

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

Property name	Data Type	Description
ElementCatalogDisplay	int	Indicates the catalog display options. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_ELCAT_STRICT (0x00) - Valid children for working from start to finish</li> <li>• Constants.FV_ELCAT_LOOSE (0x01) - Valid children for working in any order</li> <li>• Constants.FV_ELCAT_CHILDREN (0x02) - Children allowed anywhere in the parent</li> <li>• Constants.FV_ELCAT_ALL (0x03) - All elements</li> <li>• Constants.FV_ELCAT_CUSTOM (0x04) - The list of tags specified by the CustomElementList property.</li> </ul>
FirstElementDefInDoc	<a href="#">ElementDef</a>	Denotes the first element definition in the list of element definitions in the document (FO_ElementDef ID).
FirstFmtChangeListInDoc	<a href="#">FmtChangeList</a>	Specifies the ID of the first format change list in the list of format change lists in the book.(FO_FmtChangeList ID).
MaxBottomMargin	int	Specifies the maximum bottom margin allowed in the document.
MaxFirstIndent	int	Specifies the maximum first indent allowed in the document.
MaxFontSize	int	Denotes the maximum font size allowed in the document.
MaxLeading	int	Denotes the maximum leading allowed in the document.
MaxLeftIndent	int	Specifies the maximum left indent allowed in the document.
MaxLeftMargin	int	Specifies the maximum left margin allowed in the document.
MaxRightIndent	int	Specifies the maximum right indent allowed in the document.
MaxRightMargin	int	Specifies the maximum right margin allowed in the document.
MaxSpaceAbove	int	Denotes the maximum space above paragraph allowed in the document.
MaxSpaceBelow	int	Denotes the maximum space below paragraph allowed in the document.
MaxStretch	int	Denotes the maximum character stretch (set width) expressed as a percentage of normal stretch for the font (metric -10% to 1000%).
MaxTabPosition	int	Specifies the maximum tab position allowed in the document.
MaxTopMargin	int	Specifies the maximum top margin allowed in the document.
MinBottomMargin	int	Specifies the minimum bottom margin allowed in the document.
MinFirstIndent	int	Specifies the minimum first indent allowed in the document.

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

Property name	Data Type	Description
XmlEncoding	string	<p>Contains the encoding parameter of the XML Declaration for the source XML document.</p> <p>The string is empty if no encoding is specified.</p> <p>If the <code>XmlEncoding</code> property is set, the XML Declaration contains the encoding parameter with the set value on <code>Save As XML</code> operation.</p>
XmlFileEncoding	string	<p>Specifies the encoding that is detected for the source XML book. If no encoding is specified for the source XML, the value of the <code>XmlEncoding</code> property will be an empty string. In that case, if this string is set, it determines the encoding to use when saving as XML.</p> <p>If the <code>XmlEncoding</code> property has a value, the value of the <code>XmlFileEncoding</code> property may be null.</p>
XmlPublicId	string	<p>Indicates the DOCTYPE public identifier for the source XML document.</p>
XmlStandAlone	int	<p>An integer that specifies the XML standalone parameter for the XML document that is the source of the current document. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_XML_STANDALONE_YES (1)</li> <li>• Constants.FV_XML_STANDALONE_NO (2)</li> <li>• Constants.FV_XML_STANDALONE_NODEC (3)</li> <li>• Constants.FV_XML_STANDALONE_NONE (4)</li> </ul> <p>The standalone parameter is declared in the XML Declaration.</p> <p>For a file with no XML Declaration, the value is Constants.FV_XML_STANDALONE_NODEC (3).</p> <p>For an XML Declaration with no standalone parameter, this value is Constants.FV_XML_STANDALONE_NONE (4).</p>
XmlStyleSheet	string	<p>Specifies the XML stylesheet processing instruction to write out to XML when saving the document as XML. The ESTK does not verify that you use correct syntax in this string. The value you set should not include the PI delimiters, <code>&lt;?</code> and <code>?&gt;</code>. For example, the string you supply for <code>my.css</code> may be:</p> <pre>"type=\"text\\css\" href=\"my.css\""</pre> <p>Only use this string to set a specific stylesheet specification.</p> <p><b>Note:</b> The <code>GetString()</code> method always returns null for this property. To get the list of stylesheet specifications associated with a document, use the <code>XmlStyleSheetList</code> property.</p>

Property name	Data Type	Description
XmlStyleSheetList	Strings	<p>Contains a list of stylesheet processing instructions for the current document. A document can have more than one stylesheet specification associated with it. The ESTK does not verify that you use the correct syntax in these strings. The value you set should not include the PI delimiters, &lt;? and ?&gt;. For example, the string you supply for my.css may be:</p> <pre>"type=\text\css" href=\my.css"</pre> <p>Setting a list to the XmlStyleSheetList property completely overwrites the preceding list.</p>
XmlSystemId	string	Specifies the DOCTYPE system identifier for the source XML document.
XmlUseBOM	int	<p>Indicates whether a byte order mark was detected when opening the source XML. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_XML_USEBOM_YES (1)</li> <li>• Constants.FV_XML_USEBOM_NO (2)</li> </ul> <p>If this value is set to Constants.FV_XML_USEBOM_YES (1), FrameMaker writes a byte order mark in the resulting XML, when saving as XML.</p>
XmlVersion	string	<p>Indicates the XML Version that was specified in the XML Declaration when the file was opened.</p> <p>If no XML version was specified, the GetString() method returns an empty string. If this string contains an invalid XML declaration, a parsing error occurs when the document is saved as XML.</p>
XmlWellFormed	int	<p>Indicates whether the source XML qualified as well formed. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_XML_WELLFORMED_YES (1)</li> <li>• Constants.FV_XML_WELLFORMED_NO (2)</li> </ul>
TblFnCellPosition	int	<p>Denotes the placement of footnote number in footnote text. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_FN_POS_SUPER (0x00) - Superscript</li> <li>• Constants.FV_FN_POS_BASELINE (0x01) - Baseline</li> <li>• Constants.FV_FN_POS_SUB (0x02) - Subscript</li> </ul>
TblFnCellPrefix	string	Specifies the prefix to appear before the table footnote number in table cell.
TblFnCellSuffix	string	Specifies the suffix to appear after the table footnote number in table cell.
TblFnCustNumString	string	Specifies the characters for custom table footnote numbers.
TblFnFmt	string	Represents the paragraph tag of the table footnote.

Property name	Data Type	Description
TblFnNumStyle	int	Denotes the footnote numbering style for tables in the document. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_FN_NUM_NUMERIC (0x00) - Arabic</li> <li>• Constants.FV_FN_NUM_ROMAN_UC (0x01) - Roman uppercase</li> <li>• Constants.FV_FN_NUM_ROMAN_LC (0x02) - Roman lowercase</li> <li>• Constants.FV_FN_NUM_ALPHA_UC (0x03) - Alphabetic uppercase</li> <li>• Constants.FV_FN_NUM_ALPHA_LC (0x04) - Alphabetic lowercase</li> <li>• Constants.FV_FN_NUM_KANJI (0x06) - Kanji characters</li> <li>• Constants.FV_FN_NUM_ZENKAKU (0x07) - Zenkaku</li> <li>• Constants.FV_FN_NUM_ZENKAKU_UC (0x08) - Zenkaku uppercase</li> <li>• Constants.FV_FN_NUM_ZENKAKU_LC (0x09) - Zenkaku lowercase</li> <li>• Constants.FV_FN_NUM_KANJI_KAZU (0x0A) - Kazu</li> <li>• Constants.FV_FN_NUM_DAIJI (0x0B) - Daiji</li> <li>• Constants.FV_FN_NUM_CUSTOM (0x05) - Custom numbering.</li> </ul>
TblFnPosition	int	Denotes the placement of the footnote number in the text. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_FN_POS_SUPER (0x00) - Superscript</li> <li>• Constants.FV_FN_POS_BASELINE (0x01) - Baseline</li> <li>• Constants.FV_FN_POS_SUB (0x02) - Subscript</li> </ul>
TblFnPrefix	string	Indicates the prefix to appear before the number in the table footnote.
TblFnSuffix	string	Indicates the suffix to appear after the number in the table footnote.
Capitalization	int	Indicates the type of capitalization. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_CAPITAL_CASE_NORM (0)</li> <li>• Constants.FV_CAPITAL_CASE_SMALL (1)</li> <li>• Constants.FV_CAPITAL_CASE_LOWER (2)</li> <li>• Constants.FV_CAPITAL_CASE_UPPER (3)</li> </ul>
ChangeBar	int	Returns 1 if Changebars are enabled. Else, returns 0.
CharTag	string	Indicates the name of the character format tag.
Color	Color	Denotes the Spot color (FO_Color ID).



Property name	Data Type	Description
CondFmtsShown	int	Returns 1 if the document conditions are shown. Else, returns 0.
CombinedFontFamily	CombinedFontDefn	Contains the Combined font definition (FO_CombinedFontDefn)
FontEncodingName	string	Specifies the font encoding.
FontAngle	int	Denotes the font angle (specifies an index into the array of font angles provided by the session property, FontAngleNames).
FontFamily	int	Denotes the font family (specifies an index into the array of font families provided by the session property, FontFamilyNames).
FontPlatformName	string	Indicates the name that uniquely identifies a font on a specific platform.
FontPostScriptName	string	Indicates the name given to a font when it is sent to a PostScript printer.
FontSize	int	Specifies the font size (2 point to 400 point).
FontVariation	int	Denotes the font variation (specifies an index into the array of font variations provided by the session property FontVariationNames).
FontWeight	int	Denotes the font weight (specifies an index into the array of font weights provided by the session property FontWeightNames).
InCond	Objects	Indicates the condition tags that apply to the text (an array of FO_CondFmt IDs).
KernX	int	Specifies the horizontal kern value for manual kerning expressed as a percentage of an em (metric – 100% to 1000%). A positive value moves a character right and a negative value moves a character left.
KernY	int	Specifies the vertical kern value for manual kerning expressed as a percentage of an em (metric – 100% to 1000%). A positive value moves characters up and a negative value moves characters down.
Overline	int	Returns 1 if Overline style is enabled, else returns 0.
PairKern	int	Returns 1 if Pair Kern is enabled, else returns 0.
Position	int	Denotes the text position relative to the baseline of text. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_POS_NORM (0) - Normal</li> <li>Constants.FV_POS_SUPER (1) - Superscript</li> <li>Constants.FV_POS_SUB (2) - Subscript</li> </ul>
SepOverride	Color	Specifies the custom color separation override (FO_Color ID).
Stretch	int	Denotes the character stretch (set width) expressed as a percentage of normal stretch for the font (metric –10% to 1000%).
Strikethrough	int	Returns 1 if Strikethrough style is enabled. Else, returns 0.

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

Property name	Data Type	Description
ScreenWidth	int	Denotes the width of the document window in pixels.
ScreenX	int	Indicates the offset of the document window in pixels from the left side of the screen (or the left of the FrameMaker application window).  If you set a value that results in the document window being off the screen, that value is ignored and the old value is retained.
ScreenY	int	Indicates the offset of the document window in pixels from the top of the screen (or the top of the FrameMaker application window).  If you set a value that results in the document window being off the screen, that value is ignored and the old value is retained.
SnapAngle	int	Specifies the angle of rotation for Snap Rotate.
SnapGridUnits	int	Denotes the units for Snap Grid spacing. The permissible range of values is from 0 to 32768 points.
SpotColorView	int	Denotes the spot color separation view. The values range from 0 to 6.0 specifies View 1, 1 specifies View 2, and so on.
ViewBorders	int	Returns 1 if Borders is enabled, else returns 0.
ViewDisplayUnits	int	Denotes the MetricT equivalent of one unit in the current Display Units. For example, if Display Units is points, this parameter returns 65536.
ViewFontSizeUnits	int	Denotes the MetricT equivalent of one unit in the current Font Size Unit. Font size units can be either Points or Q. If Points, this returns 65536. If Q, this returns 47098.
ViewGrid	int	Returns 1 if View Grid is enabled. Else, returns 0.
ViewGridUnits	int	Denotes the units for Grid Lines.
ViewNoGraphics	int	Returns 1 if Graphics is not enabled. Else, set to 0.
ViewPageScrolling	int	Denotes the Page scrolling technique. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_SCROLL_VARIABLE (0)</li> <li>• Constants.FV_SCROLL_HORIZONTAL (1)</li> <li>• Constants.FV_SCROLL_VERTICAL (2)</li> <li>• Constants.FV_SCROLL_FACING (3)</li> </ul>
ViewRulers	int	Returns 1 if Rulers is enabled. Else, returns 0.
ViewRulerUnits	int	Denotes the units to display rulers.
ViewTextSymbols	int	Returns 1 if Text Symbols is enabled. Else, returns 0.
Zoom	int	Specifies the zoom percentage of document (metric 25% to 1600%).
DocFluidFlow	Flow	Specifies the flow to set to fluid view. To turn this off, set the value of this property to 0.
ViewOnlyDeadCodes	UInts	Indicates the F-codes that cannot be executed in the document.

Property name	Data Type	Description
ViewOnlySelect	int	Specifies whether user can select text or graphics in the document. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_VOS_USER_ONLY (1) - the user can select text when pressing modifier keys, and link targets (cross-reference sources and newliniks) do not highlight.</li> <li>Constants.FV_VOS_NONE (2) - the user cannot select text, and links targets do not highlight.</li> <li>Constants.FV_VOS_YES(3) - the user can select text (using modifier keys) and link targets are highlighted.</li> </ul>
ViewOnlyWinBorders	int	Returns 1 if the document has normal document borders; returns 0 if the document scroll bars and border buttons are suppressed.
ViewOnlyWinMenubar	int	Returns 1 if the document has a document window menu bar. Else, returns 0.
ViewOnlyWinPalette	int	Returns 1 if the document is a palette. Else, returns 0.
ViewOnlyWinPopup	int	Returns 1 if the document window pop-up menu is available. Else, returns 0.
ViewOnlyXRef	int	Specifies the behavior of cross-references in the document. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_VOX_NOT_ACTIVE(0) - Cross references are not active.</li> <li>Constants.FV_VOX_GOTO_BEHAVIOR(1) - Internal cross-references are active.</li> <li>Constants.FV_VOX_OPEN_BEHAVIOR(2) - External cross-references are active.</li> <li>Constants.FV_VOX_ALERT(3) - Alert appears when cross-reference is clicked.</li> </ul>
PDFGenerateForReview	int	This flag is used to generate a PDF for review workflow that is getting review comments in PDF and importing them back to FrameMaker document. It helps in importing back the comments correctly. When this flag is turned ON, the behavior is same as if the "Generate for review only" checkbox is ON in the PDF setup dialog.
PDFViewPDF	int	Set to 1 to open the PDF after creating it with the Save as PDF command.
StructuredApplicationForOpen	string	Specifies the application that has been used to open the document.
SecNumComputeMethod	int	Denotes the type of numbering applied on a particular section of a document. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_NUM_CONTINUE (0x01)</li> <li>Constants.FV_NUM_RESTART (0x02)</li> <li>Constants.FV_NUM_SAME (0x03)</li> </ul>

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

Property name	Data Type	Description
SubsectionNumStyle	int	Denotes the subsection numbering style. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PAGE_NUM_NUMERIC(0x00) - Arabic</li> <li>Constants.FV_PAGE_NUM_ROMAN_UC(0x01) - Roman uppercase</li> <li>Constants.FV_PAGE_NUM_ROMAN_LC(0x02) - Roman lowercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_UC(0x03) - Alphabetic uppercase</li> <li>Constants.FV_PAGE_NUM_ALPHA_LC(0x04) - Alphabetic lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI(0x05) - Kanji characters</li> <li>Constants.FV_PAGE_NUM_ZENKAKU(0x06) - Zenkaku</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_UC(0x07) - Zenkaku uppercase</li> <li>Constants.FV_PAGE_NUM_ZENKAKU_LC(0x08) - Zenkaku lowercase</li> <li>Constants.FV_PAGE_NUM_KANJI_KAZU(0x09) - Kazu</li> <li>Constants.FV_PAGE_NUM_DAIJI(0x0a) - Daiji</li> </ul>
SubsectionNumber	int	Denotes the subsection number in the document
SubsectionNumText	string	Indicates the text to use for the <code>SubsectionNumber</code> parameter.
TrackChangesOn	bool	Set to True if Track Changes are on in the document. Else, set to False.
PreviewState	int	Sets/retrieves the state of Track Text Edits of the document. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PREVIEW_OFF_TRACK_CHANGE (0) - Displays text edits using the defined styles.</li> <li>Constants.FV_PREVIEW_ON_ORIGINAL (1)- Preview of the document as if no text edits were done.</li> <li>Constants.FV_PREVIEW_ON_FINAL (2) - Preview of the document as if all the text edits were incorporated.</li> </ul>
TrackChangesAddedColor	Color	Sets/retrieves the color in which tracked additions are displayed.
TrackChangesDeletedColor	Color	Sets/retrieves the color in which tracked deletions are displayed.
ReviewerNameList	Strings	Retrieves the list of reviewers of a document.
FirstAttrCondExprInDoc	AttrCondExpr	Denotes the first Attribute Conditional Expression in the list of attribute conditional expressions in the document (FO_AttrCondExpr).

Property name	Data Type	Description
BooleanConditionExpression	string	Denotes the actual conditional expression used in the document.
BooleanConditionState	int	Set to 1 to indicate that the conditional expression defined by the <code>BooleanConditionExpression</code> parameter is applied to the document.  Set to 0 to indicate that the condition is not applied to the document.
BooleanConditionExpressionTag	string	Indicates the tag name of the expression applied in the document.
ShowElementDescriptiveNames	bool	Set to True to show element descriptive names in the element catalog, as specified in element definition.
UseInitialStructureOfAutoInsertedElements	bool	If set to True, auto-insertion rules will be processed recursively. For example: If an element is inserted automatically, and if auto-insertion rules exist for this element in the element definition, then those rules are also processed.

## Doc methods

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

# Element

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



Property name	Data Type	Description
ElementType	int	The type of element. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_FO_CONTAINER (1)</li> <li>• Constants.FV_FO_SYS_VAR (2)</li> <li>• Constants.FV_FO_XREF (3)</li> <li>• Constants.FV_FO_MARKER (4)</li> <li>• Constants.FV_FO_FOOTNOTE (5)</li> <li>• Constants.FV_FO_GRAPHIC (6)</li> <li>• Constants.FV_FO_EQN (7)</li> <li>• Constants.FV_FO_TBL (8)</li> <li>• Constants.FV_FO_TBL_TITLE (9)</li> <li>• Constants.FV_FO_TBL_HEADING (10)</li> <li>• Constants.FV_FO_TBL_BODY (11)</li> <li>• Constants.FV_FO_TBL_FOOTING (12)</li> <li>• Constants.FV_FO_TBL_ROW (13)</li> <li>• Constants.FV_FO_TBL_CELL (14)</li> <li>• Constants.FV_FO_RUBIGROUP (15)</li> <li>• Constants.FV_FO_RUBI (16)</li> </ul>
MatchingFirstPgfClauses	Objects	IDs of the first paragraph clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
FormatOverride	int	Returns 1 if the element has a format override. Else, returns 0.
ElementMarkedForNamedDestination	int	Used for generatig PDF. If 1, this element will have a corresponding <i>Named Destination</i> in the generated PDF.
MatchingLastPgfClauses	Objects	IDs of the last paragraph clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
MatchingObjectClauses	Objects	IDs of the object clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
MatchingPrefixClauses	Objects	IDs of the prefix clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
MatchingSuffixClauses	Objects	IDs of the suffix clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
MatchingTextClauses	Objects	IDs of the text clauses (FO_FmtRuleClause IDs) in the element's definition that apply to the element.
TextRange	<a href="#">TextRange</a>	Denotes the Text range that the element encompasses (see the explanation below).
TopicRefAbsolutePath	string	Denotes the absolute file path for the topic reference
Unique	int	Unique ID of the element.

Property name	Data Type	Description
UserString	string	A string to which clients can store private data.
BookComponent	<a href="#">BookComponent</a>	Component file in book (FO_BookComponent ID).
ElementDef	<a href="#">ElementDef</a>	Element's element definition (FO_ElementDef ID).
FirstChildElement	<a href="#">Element</a>	Denote the element's first child element (FO_Element ID) if the element is a container, .
LastChildElement	<a href="#">Element</a>	Denotes the element's last child element (FO_Element ID) if the element is a container.
NextSiblingElement	<a href="#">Element</a>	Element's next sibling element (FO_Element ID).
NextDITAConrefElementInDoc	string	The next Conref element in the document
NextDITAXrefElementInDoc	string	The next XRef element in the document
NextDITALinkElementInDoc	string	The next Link element in the document
NextDITATopicrefElementInDoc	string	The next Topicref element in the document
NextDITATopicsetrefElementInDoc	string	The next Topicsetref element in the document
Object	FMOBJECT	<p>ID of the object that an element contains. The type of object the ID specifies depends on the element definition as follows:</p> <ul style="list-style-type: none"> <li>• Constants.FV_FO_SYS_VAR (2): FO_Var</li> <li>• Constants.FV_FO_XREF (3): FO_XRef</li> <li>• Constants.FV_FO_MARKER (4): FO_Marker</li> <li>• Constants.FV_FO_FOOTNOTE (5): FO_Fn</li> <li>• Constants.FV_FO_GRAPHIC (6): FO_AFrame (containing the graphic)</li> <li>• Constants.FV_FO_EQN (7): FO_AFrame (containing the equation)</li> <li>• Constants.FV_FO_TBL (8): FO_Tbl</li> <li>• Constants.FV_FO_TBL_TITLE (9): FO_Tbl</li> <li>• Constants.FV_FO_TBL_HEADING (10): FO_Tbl</li> <li>• Constants.FV_FO_TBL_BODY (11): FO_Tbl</li> <li>• Constants.FV_FO_TBL_FOOTING (12): FO_Tbl</li> <li>• Constants.FV_FO_TBL_ROW (13): FO_Row</li> <li>• Constants.FV_FO_TBL_CELL (14): FO_Cell</li> <li>• Constants.FV_FO_RUBIGROUP (15): FO_Rubi</li> <li>• Constants.FV_FO_RUBI (16): FO_Rubi</li> </ul>
ParentElement	<a href="#">Element</a>	Element's parent element (FO_Element ID).
PrevSiblingElement	<a href="#">Element</a>	Element's previous sibling element (FO_Element ID).

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

Property name	Data Type	Description
NextInvalidElement	<a href="#">Element</a>	Next invalid element in the document (FO_Element ID).
TextIsInvalidInElement	int	1 if the element contains only text and the element definition disallows it.  TextIsInvalidInElement is obsolete and is no longer supported.
ValidationFlags	int	<p>Bit flags specifying the element's validity.</p> <p>To determine all the ways in which an element is invalid without querying all the validation properties, query the ValidationFalgs property.</p> <p>Each bit flag in the returned value represents the value of the validation property with the corresponding name.</p> <p>For example, if the <code>ElementTypeMismatch</code> property is <code>True</code>, the <code>Constants.FV_ELEM_TYPE_MISMATCH (0x0002)</code> flag is set.</p> <p>The various bit flags are as follows:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_ELEM_UNDEFINED (0x0001)</code></li> <li>• <code>Constants.FV_ELEM_TYPE_MISMATCH (0x0002)</code></li> <li>• <code>Constants.FV_ELEM_EXCLUDED (0x0004)</code></li> <li>• <code>Constants.FV_ELEM_INVALID_IN_PARENT (0x0008)</code></li> <li>• <code>Constants.FV_ELEM_INVALID_AT_POSITION (0x0010)</code></li> <li>• <code>Constants.FV_ELEM_HAS_TEXT_INVALID (0x0020)</code></li> <li>• <code>Constants.FV_ELEM_CONTENT_MUST_BE_EMPTY (0x0040)</code></li> <li>• <code>Constants.FV_ELEM_MISSING_CONTENT_BEFORE (0x0080)</code></li> <li>• <code>Constants.FV_ELEM_MISSING_CONTENT_AT_BEG (0x0100)</code></li> <li>• <code>Constants.FV_ELEM_MISSING_CONTENT_AT_END (0x0200)</code></li> <li>• <code>Constants.FV_ELEM_NOT_VALID_AS_ROOT (0x0400)</code></li> <li>• <code>Constants.FV_ELEM_BOOK_COMP_MISSING (0x0800)</code></li> <li>• <code>Constants.FV_ELEM_BOOK_COMP_INVALID (0x1000)</code></li> <li>• <code>Constants.FV_ELEM_ATTRVAL_REQUIRED (0x2000)</code></li> <li>• <code>Constants.FV_ELEM_ATTRVAL_INVALID (0x4000)</code></li> <li>• <code>Constants.FV_ELEM_CONTENT_STRICTLY_VALID (0x10000)</code></li> <li>• <code>Constants.FV_ELEM_CONTENT_LOOSELY_VALID (0x20000)</code></li> </ul>

## Element methods

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

## ElementCatalogEntries

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

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

### ElementCatalogEntries methods

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

## ElementCatalogEntry

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

### ElementCatalogEntry methods

[ElementCatalogEntry](#).

## ElementDef

Property name	Data Type	Description
AttributeDefs	<a href="#">AttributeDefs</a>	The element definition's attribute definitions.
BannerText	string	Property of element definition object, FO_ElementDef. Users can set, change or query the banner text associated with an element definition object.
Comment	string	Text string of the comment.

Property name	Data Type	Description
InitStructurePattern	string	The initial structure pattern; for table elements, a comma delimited string that specifies the necessary child elements to automatically insert.
ElementDefType	int	Type of formatter object represented by the element with element definition. Constants.FV_FO_CONTAINER (1) identifies a container element. Other values identify object (non-container) elements.  The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_FO_UNSPECIFIED (0)</li> <li>• Constants.FV_FO_CONTAINER (1)</li> <li>• Constants.FV_FO_SYS_VAR (2)</li> <li>• Constants.FV_FO_XREF (3)</li> <li>• Constants.FV_FO_MARKER (4)</li> <li>• Constants.FV_FO_FOOTNOTE (5)</li> <li>• Constants.FV_FO_GRAPHIC (6)</li> <li>• Constants.FV_FO_EQN (7)</li> <li>• Constants.FV_FO_TBL (8)</li> <li>• Constants.FV_FO_TBL_TITLE (9)</li> <li>• Constants.FV_FO_TBL_HEADING (10)</li> <li>• Constants.FV_FO_TBL_BODY (11)</li> <li>• Constants.FV_FO_TBL_FOOTING (12)</li> <li>• Constants.FV_FO_TBL_ROW (13)</li> <li>• Constants.FV_FO_TBL_CELL (14)</li> </ul>
ElementInCatalog	int	1 if the element is in the Element Catalog.
ElementPgFormat	string	The name of the paragraph format applied to the element.
Exclusions	Strings	List of excluded elements.
FirstPgRules	Objects	The IDs of the first paragraph format rules (FO_FmtRule IDs).  <i>Note: To set the format rules for an element definition, use NewFmtRuleObject ().</i>
GeneralRule	string	Text of the element's general rule.
GeneralRuleErrorOffsets	Ints	Contains the error offsets (two positions are specified only if the content rule is ambiguous).
Inclusions	Strings	List of included elements.
LastPgRules	Objects	The IDs of the last paragraph format rules (FO_FmtRule IDs).
Name	string	Name of the element definition.

Property name	Data Type	Description
NextElementDefInDoc	<a href="#">ElementDef</a>	Next element definition in the document's list of element definitions (FO_ElementDef ID).
ObjectFmtRules	Objects	The IDs of the object format rules (FO_FmtRule IDs).
PrefixRules	Objects	The IDs of the prefix format rules (FO_FmtRule IDs).
SuffixRules	Objects	The IDs of the suffix format rules (FO_FmtRule IDs).
TextFmtRules	Objects	The IDs of the text format rules (FO_FmtRule IDs).
ValidHighestLevel	int	1 if the element can be used as the highest-level element for a flow.
ElementDataType	int	Retrieves/sets the data type of Element. The possible values are one of: <ul style="list-style-type: none"> <li>Constants.FV_DT_UNSPECIFIED(0)</li> <li>Constants.FV_DT_INTEGER(1)</li> <li>Constants.FV_DT_FLOAT(2)</li> </ul>
AllowedDataRange	TBD	Defines the permissible range of values for the values defined in the <code>ElementDataType</code> property.
DescriptiveTag	string	A small description about the element. If the user specifies this in the EDD, then the element description can be viewed in the Element Catalog of the structured document.
ElementDescription	string	Reserved for future use.
AlsoInsert	<a href="#">Strings</a>	The list of the tags of child elements that are automatically inserted when an element is initially added.
AlsoInserts	<a href="#">TypedVal</a>	The list of list of the tags of child elements that are automatically inserted when an element is initially added.  For example: Setting the <code>AlsoInserts</code> property of element definition of A to <code>[[a1, a11],[a2, a21]]</code> , will automatically insert all child elements of A [a1, a2, a11 and a21] when A is inserted.

## ElementDef methods

[Delete](#), [ElementDefIsText](#), [GetAttributeDefs](#), [GetProps](#), [NewElement](#), [NewElementInHierarchy](#), [NewFirstPgfRules](#), [NewLastPgfRules](#), [NewObjectFmtRules](#), [NewPrefixRules](#), [NewSuffixRules](#), [NewTextFmtRules](#), [ObjectValid](#), [SetProps](#), [WrapElement](#).

## ElementLoc

Property name	Data Type	Description
parent	<a href="#">Element</a>	ID of the parent element.
child	<a href="#">Element</a>	ID of the child element.
offset	int	Offset within the parent or child element.

## ElementLoc methods

[ElementLoc](#).

## ElementRange

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

## ElementRange methods

[ElementRange](#).

## Ellipse

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



Property name	Data Type	Description
Dash	Metrics	The dash style. The Dash property specifies a dash pattern that is repeated for the length of an object's border.  The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	The fill pattern (numbers between 0 and 15). The ESTK provides constants for the following fill patterns: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object (AFrame or UnAnchoredFrame).
GraphicCantBeSelected	int	1 if the graphic object cannot be selected; 0 otherwise.
GraphicsSelected	int	1 if the graphic object is selected; 0, otherwise.
GroupParent	Group	Group that the object is in (FO_Group ID). Anchored and unanchored frames do not have this property.
HeadArrow	int	1 if the line has an arrow head; otherwise, 0.
Height	int	Denotes the height of the object. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Type of line end. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Distance from the left side of the parent frame (-216 inches to 216 inches).  <i>Note: For anchored frames, the distance is automatically calculated from the left side of the page frame. You cannot manually set this property for anchored frames.</i>

Property name	Data Type	Description
LocY	int	Distance from the top of the parent frame (–216 inches to 216 inches).  <i>Note: For anchored frames, the distance is automatically calculated from the top of the page frame. You cannot manually set this property for anchored frames.</i>
NextGraphicInDoc	FObject	Next graphic object in the document.
NextGraphicInFrame	FObject	Next graphic object in the frame.
NextGraphicInGroup	FObject	Next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Next selected graphic object in document.
ObjectAttributes	Strings	A list of strings, each string expressing an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <tag>string_text</tag>. You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the objec. The value is one of:  <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	The fill pattern (numbers between 0 and 15). The ESTK provides constants for the following fill patterns:  <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Previous graphic object in the frame.
PrevGraphicInGroup	FObject	Previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object.  The value is one of:  <ul style="list-style-type: none"> <li>• Constants.FV_TR_NONE (0x01)</li> <li>• Constants.FV_TR_CONTOUR (0x02)</li> <li>• Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap if the object is a runaround object.
TailArrow	int	1 if arrowhead is enabled at the beginning of the line. 0, otherwise.

Property name	Data Type	Description
TintPercent	int	Denotes the tint percentage.
Unique	int	The graphic object's UID.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## Ellipse methods

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

## FCodes

Property name	Data Type
FCodes.NULLINPUT	NULLINPUT <b>Value -1</b>
FCodes.KBD_INPUT	KBD_INPUT <b>Value 1</b>
FCodes.START_DIALOG	START_DIALOG <b>Value 0xC100</b>
FCodes.END_DIALOG	END_DIALOG <b>Value 0xC200</b>
FCodes.START_WINDOW	START_WINDOW <b>Value 0xC400</b>
FCodes.END_WINDOW	END_WINDOW <b>Value 0xC500</b>
FCodes.LEAVE_DOC	LEAVE_DOC <b>Value 0xC600</b>
FCodes.SERVER_CANCEL	SERVER_CANCEL <b>Value 0xC001</b>
FCodes.SERVER_QUERY	SERVER_QUERY <b>Value 0xC002</b>
FCodes.MENU_BARSTART	MENU_BARSTART <b>Value 0xC301</b>
FCodes.MENU_WAITBIT	MENU_WAITBIT <b>Value 0x0010</b>
FCodes.MENU_DOWNWAIT	MENU_DOWNWAIT <b>Value 0xC312</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.MENU_RIGHTWAIT	MENU_RIGHTWAIT <b>Value</b> 0xC313
FCodes.MENU_DOWN	MENU_DOWN <b>Value</b> 0xC304
FCodes.MENU_RIGHT	MENU_RIGHT <b>Value</b> 0xC305
FCodes.MENU_DONE	MENU_DONE <b>Value</b> 0xC30F
FCodes.KBD_GBL_START	KBD_GBL_START <b>Value</b> 0x100
FCodes.CSR_HOME	CSR_HOME <b>Value</b> 0x100
FCodes.CSR_UP	CSR_UP <b>Value</b> 0x101
FCodes.CSR_DOWN	CSR_DOWN <b>Value</b> 0x102
FCodes.CSR_RIGHT	CSR_RIGHT <b>Value</b> 0x103
FCodes.CSR_LEFT	CSR_LEFT <b>Value</b> 0x104
FCodes.CSR_BOL	CSR_BOL <b>Value</b> 0x105
FCodes.CSR_EOL	CSR_EOL <b>Value</b> 0x106
FCodes.CSR_BOW	CSR_BOW <b>Value</b> 0x107
FCodes.CSR_EOW	CSR_EOW <b>Value</b> 0x108
FCodes.CSR_BOS	CSR_BOS <b>Value</b> 0x109
FCodes.CSR_EOS	CSR_EOS <b>Value</b> 0x10A
FCodes.CSR_BOP	CSR_BOP <b>Value</b> 0x10B
FCodes.CSR_EOP	CSR_EOP <b>Value</b> 0x10C

<b>Property name</b>	<b>Data Type</b>
FCodes.CSR_TOTR	CSR_TOTR <b>Value 0x10D</b>
FCodes.CSR_BOTR	CSR_BOTR <b>Value 0x10E</b>
FCodes.CSR_TOP	CSR_TOP <b>Value 0x10F</b>
FCodes.CSR_BOT	CSR_BOT <b>Value 0x110</b>
FCodes.DEL_CHARBWD	DEL_CHARBWD <b>Value 0x112</b>
FCodes.DEL_CHARFWD	DEL_CHARFWD <b>Value 0x113</b>
FCodes.DEL_BOW	DEL_BOW <b>Value 0x114</b>
FCodes.DEL_EOW	DEL_EOW <b>Value 0x115</b>
FCodes.DEL_EOL	DEL_EOL <b>Value 0x116</b>
FCodes.DEL_EOS	DEL_EOS <b>Value 0x117</b>
FCodes.DEL_SEL	DEL_SEL <b>Value 0x118</b>
FCodes.DEL_BOL	DEL_BOL <b>Value 0x119</b>
FCodes.KBD_KERNUP	KBD_KERNUP <b>Value 0x11A</b>
FCodes.KBD_KERNDOWN	KBD_KERNDOWN <b>Value 0x11B</b>
FCodes.KBD_KERNLEFT	KBD_KERNLEFT <b>Value 0x11C</b>
FCodes.KBD_KERNRIGHT	KBD_KERNRIGHT <b>Value 0x11D</b>
FCodes.KBD_KERNHOME	KBD_KERNHOME <b>Value 0x11E</b>
FCodes.KBD_KERNUP6	KBD_KERNUP6 <b>Value 0x121</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_KERNDOWN6	KBD_KERNDOWN6 <b>Value 0x122</b>
FCodes.KBD_KERNLEFT6	KBD_KERNLEFT6 <b>Value 0x123</b>
FCodes.KBD_KERNRIGHT6	KBD_KERNRIGHT6 <b>Value 0x124</b>
FCodes.KBD_OBJJUP	KBD_OBJJUP <b>Value 0x125</b>
FCodes.KBD_OBJDOWN	KBD_OBJDOWN <b>Value 0x126</b>
FCodes.KBD_OBJLEFT	KBD_OBJLEFT <b>Value 0x127</b>
FCodes.KBD_OBJRIGHT	KBD_OBJRIGHT <b>Value 0x128</b>
FCodes.CSR_NEXT_BOW	CSR_NEXT_BOW <b>Value 0x140</b>
FCodes.CSR_NEXT_BOS	CSR_NEXT_BOS <b>Value 0x141</b>
FCodes.CSR_NEXT_BOP	CSR_NEXT_BOP <b>Value 0x142</b>
FCodes.CSR_FIRST_COL	CSR_FIRST_COL <b>Value 0x143</b>
FCodes.CSR_NEXT_COL	CSR_NEXT_COL <b>Value 0x144</b>
FCodes.CSR_NEXT_ELEMENT	CSR_NEXT_ELEMENT <b>Value 0x145</b>
FCodes.CSR_PREV_ELEMENT	CSR_PREV_ELEMENT <b>Value 0x146</b>
FCodes.CSR_BOE	CSR_BOE <b>Value 0x149</b>
FCodes.CSR_EOE	CSR_EOE <b>Value 0x14A</b>
FCodes.CSR_BEFORE_ELEMENT	CSR_BEFORE_ELEMENT <b>Value 0x14B</b>
FCodes.CSR_INTO_CHILD	CSR_INTO_CHILD <b>Value 0x14D</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.CSR_PREV_EOS	CSR_PREV_EOS <b>Value 0x937</b>
FCodes.CSR_PREV_EOW	CSR_PREV_EOW <b>Value 0x938</b>
FCodes.DEL_WORD_START	DEL_WORD_START <b>Value 0x160</b>
FCodes.DEL_WORD_END	DEL_WORD_END <b>Value 0x161</b>
FCodes.DEL_NEXT_SS	DEL_NEXT_SS <b>Value 0x162</b>
FCodes.DEL_BOS	DEL_BOS <b>Value 0x163</b>
FCodes.KBD_NUMLOCK	KBD_NUMLOCK <b>Value 0x170</b>
FCodes.KBD_GBL_END	KBD_GBL_END <b>Value 0x1FF</b>
FCodes.KBD_BACKTAB	KBD_BACKTAB <b>Value 0x220</b>
FCodes.KBD_SHFTSPACE	KBD_SHFTSPACE <b>Value 0x221</b>
FCodes.KBD_FIRSTTAB	KBD_FIRSTTAB <b>Value 0x222</b>
FCodes.KBD_TABLE_TAB	KBD_TABLE_TAB <b>Value 0x223</b>
FCodes.KBD_TAB	KBD_TAB <b>Value 0x219</b>
FCodes.KBD_XCHARS	KBD_XCHARS <b>Value 0x224</b>
FCodes.KBD_SOFTHYPHEN	KBD_SOFTHYPHEN <b>Value 0x225</b>
FCodes.KBD_DONTHYPHEN	KBD_DONTHYPHEN <b>Value 0x226</b>
FCodes.KBD_HARDHYPHEN	KBD_HARDHYPHEN <b>Value 0x227</b>
FCodes.KBD_HARDSPACE	KBD_HARDSPACE <b>Value 0x228</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_HARDRETURN	KBD_HARDRETURN <b>Value</b> 0x229
FCodes.KBD_NUMSPACE	KBD_NUMSPACE <b>Value</b> 0x22A
FCodes.KBD_THinspace	KBD_THinspace <b>Value</b> 0x22B
FCodes.KBD_EMSPACE	KBD_EMSPACE <b>Value</b> 0x22C
FCodes.KBD_ENSPACE	KBD_ENSPACE <b>Value</b> 0x22D
FCodes.KBD_OPENLINE	KBD_OPENLINE <b>Value</b> 0x22E
FCodes.KBD_RETURN	KBD_RETURN <b>Value</b> 0x22F
FCodes.KBD_FPREV	KBD_FPREV <b>Value</b> 0x230
FCodes.KBD_FNEXT	KBD_FNEXT <b>Value</b> 0x231
FCodes.KBD_RONCE	KBD_RONCE <b>Value</b> 0x232
FCodes.KBD_RGLOBAL	KBD_RGLOBAL <b>Value</b> 0x233
FCodes.KBD_RANDF	KBD_RANDF <b>Value</b> 0x234
FCodes.KBD_SETSEARCH	KBD_SETSEARCH <b>Value</b> 0x235
FCodes.HIGH_CHAR	HIGH_CHAR <b>Value</b> 0x240
FCodes.HIGH_WORD	HIGH_WORD <b>Value</b> 0x241
FCodes.HIGH_LINE	HIGH_LINE <b>Value</b> 0x242
FCodes.HIGH_SENT	HIGH_SENT <b>Value</b> 0x243
FCodes.HIGH_PGF	HIGH_PGF <b>Value</b> 0x244



<b>Property name</b>	<b>Data Type</b>
FCodes.HIGH_SHL	HIGH_SHL <b>Value</b> 0x245
FCodes.HIGH_SHR	HIGH_SHR <b>Value</b> 0x246
FCodes.HIGH_CLEAR	HIGH_CLEAR <b>Value</b> 0x247
FCodes.HIGH_FLOW	HIGH_FLOW <b>Value</b> 0x248
FCodes.HIGH_ELEMENT	HIGH_ELEMENT <b>Value</b> 0x249
FCodes.KBD_ABORT	KBD_ABORT <b>Value</b> 0x250
FCodes.KBD_CAPTURE	KBD_CAPTURE <b>Value</b> 0x251
FCodes.KBD_ECAPTURE	KBD_ECAPTURE <b>Value</b> 0x256
FCodes.KBD_RECORD	KBD_RECORD <b>Value</b> 0x252
FCodes.KBD_GETTRIGGER	KBD_GETTRIGGER <b>Value</b> 0x253
FCodes.KBD_ABORT_DLGS	KBD_ABORT_DLGS <b>Value</b> 0x254
FCodes.KBD_WAIT_DLGS	KBD_WAIT_DLGS <b>Value</b> 0x255
FCodes.TXT_BOLD	TXT_BOLD <b>Value</b> 0x260
FCodes.TXT_ITALIC	TXT_ITALIC <b>Value</b> 0x261
FCodes.TXT_UNDERLINE	TXT_UNDERLINE <b>Value</b> 0x262
FCodes.TXT_PLAIN	TXT_PLAIN <b>Value</b> 0x263
FCodes.TXT_SUPER	TXT_SUPER <b>Value</b> 0x264
FCodes.TXT_SUB	TXT_SUB <b>Value</b> 0x265

<b>Property name</b>	<b>Data Type</b>
FCodes.TXT_NORMAL	TXT_NORMAL <b>Value</b> 0x266
FCodes.TXT_INCSIZE	TXT_INCSIZE <b>Value</b> 0x267
FCodes.TXT_DECSIZE	TXT_DECSIZE <b>Value</b> 0x268
FCodes.TXT_SELINCSIZE	TXT_SELINCSIZE <b>Value</b> 0x11F
FCodes.TXT_SELDECSIZE	TXT_SELDECSIZE <b>Value</b> 0x120
FCodes.TXT_SQUEEZE	TXT_SQUEEZE <b>Value</b> 0x269
FCodes.TXT_SPREAD	TXT_SPREAD <b>Value</b> 0x26A
FCodes.TXT_NOSTRETCH	TXT_NOSTRETCH <b>Value</b> 0x130
FCodes.TXT_LESSSTRETCH	TXT_LESSSTRETCH <b>Value</b> 0x131
FCodes.TXT_MORESTRETCH	TXT_MORESTRETCH <b>Value</b> 0x132
FCodes.TXT_BAM	TXT_BAM <b>Value</b> 0x26B
FCodes.TXT_STRIKEOUT	TXT_STRIKEOUT <b>Value</b> 0x26C
FCodes.TXT_DEFAULT	TXT_DEFAULT <b>Value</b> 0x26D
FCodes.TXT_OVERLINE	TXT_OVERLINE <b>Value</b> 0x26E
FCodes.TXT_CHANGEBAR	TXT_CHANGEBAR <b>Value</b> 0x26F
FCodes.TXT_KERN	TXT_KERN <b>Value</b> 0x270
FCodes.TXT_OUTLINE	TXT_OUTLINE <b>Value</b> 0x271
FCodes.TXT_SHADOW	TXT_SHADOW <b>Value</b> 0x272

<b>Property name</b>	<b>Data Type</b>
FCodes.TXT_MINICAPS	TXT_MINICAPS <b>Value</b> 0x273
FCodes.TXT_INITCAPS	TXT_INITCAPS <b>Value</b> 0x274
FCodes.TXT_UPPERCASE	TXT_UPPERCASE <b>Value</b> 0x275
FCodes.TXT_LOWERCASE	TXT_LOWERCASE <b>Value</b> 0x276
FCodes.TXT_NORMALCASE	TXT_NORMALCASE <b>Value</b> 0x277
FCodes.TXT_DBLUNDERLINE	TXT_DBLUNDERLINE <b>Value</b> 0x278
FCodes.TXT_NUMUNDERLINE	TXT_NUMUNDERLINE <b>Value</b> 0x279
FCodes.TXT_TSUME	TXT_TSUME <b>Value</b> 0x2A5
FCodes.TXT_SELBOLD	TXT_SELBOLD <b>Value</b> 0x27A
FCodes.TXT_SELITALIC	TXT_SELITALIC <b>Value</b> 0x27B
FCodes.TXT_SELUNDERLINE	TXT_SELUNDERLINE <b>Value</b> 0x27C
FCodes.TXT_SELPLAIN	TXT_SELPLAIN <b>Value</b> 0x27D
FCodes.TXT_7	TXT_7 <b>Value</b> 0xC00
FCodes.TXT_9	TXT_9 <b>Value</b> 0xC01
FCodes.TXT_10	TXT_10 <b>Value</b> 0xC02
FCodes.TXT_12	TXT_12 <b>Value</b> 0xC03
FCodes.TXT_14	TXT_14 <b>Value</b> 0xC04
FCodes.TXT_18	TXT_18 <b>Value</b> 0xC05

<b>Property name</b>	<b>Data Type</b>
FCodes.TXT_24	TXT_24 <b>Value 0xC06</b>
FCodes.TXT_USIZE	TXT_USIZE <b>Value 0xC07</b>
FCodes.TXT_OTHERSIZE	TXT_OTHERSIZE <b>Value 0xC08</b>
FCodes.TXT_FAMILY_AND_VARIATION	TXT_FAMILY_AND_VARIATION <b>Value 0xC09</b>
FCodes.TXT_FAMILY_AND_VARIATION_PLATFORM	TXT_FAMILY_AND_VARIATION_PLATFORM <b>Value 0xC0A</b>
FCodes.PGF_APPLY_TAG	PGF_APPLY_TAG <b>Value 0xC10</b>
FCodes.CHAR_APPLY_TAG	CHAR_APPLY_TAG <b>Value 0xC11</b>
FCodes.MENU_EXPOSE_WIN	MENU_EXPOSE_WIN <b>Value 0xC12</b>
FCodes.MRU_SELECT_FILE	MRU_SELECT_FILE <b>Value 0xC13</b>
FCodes.MENU_IMPORT_FILE	MENU_IMPORT_FILE <b>Value 0xC14</b>
FCodes.KBD_EXPORT_GRAPHIC	KBD_EXPORT_GRAPHIC <b>Value 0xC15</b>
FCodes.KBD_EXPORT_DOCUMENT	KBD_EXPORT_DOCUMENT <b>Value 0xC16</b>
FCodes.TXT_FONT	TXT_FONT <b>Value 0xC17</b>
FCodes.PGF_APPLY_CAT_TO_SEL	PGF_APPLY_CAT_TO_SEL <b>Value 0xF3D</b>
FCodes.CHAR_APPLY_CAT_TO_SEL	CHAR_APPLY_CAT_TO_SEL <b>Value 0xF3E</b>
FCodes.PGF_HYPHENON	PGF_HYPHENON <b>Value 0x27E</b>
FCodes.PGF_HYPHENOFF	PGF_HYPHENOFF <b>Value 0x27F</b>
FCodes.PGF_INCLINE	PGF_INCLINE <b>Value 0x280</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.PGF_DECLINE	PGF_DECLINE <b>Value</b> 0x281
FCodes.PGF_CENTER	PGF_CENTER <b>Value</b> 0x282
FCodes.PGF_LEFT	PGF_LEFT <b>Value</b> 0x283
FCodes.PGF_RIGHT	PGF_RIGHT <b>Value</b> 0x284
FCodes.PGF_FULL	PGF_FULL <b>Value</b> 0x285
FCodes.PGF_BAM	PGF_BAM <b>Value</b> 0x286
FCodes.PGF_LINEFIX	PGF_LINEFIX <b>Value</b> 0x287
FCodes.PGF_LINEFLOAT	PGF_LINEFLOAT <b>Value</b> 0x288
FCodes.PGF_UNIFY	PGF_UNIFY <b>Value</b> 0x289
FCodes.PGF_PAGETOP	PGF_PAGETOP <b>Value</b> 0x28A
FCodes.PGF_COLTOP	PGF_COLTOP <b>Value</b> 0x28B
FCodes.PGF_LEFTTOP	PGF_LEFTTOP <b>Value</b> 0x28C
FCodes.PGF_RIGHTTOP	PGF_RIGHTTOP <b>Value</b> 0x28D
FCodes.PGF_ANYPLACE	PGF_ANYPLACE <b>Value</b> 0x28E
FCodes.KBD_ALIGN_TOP	KBD_ALIGN_TOP <b>Value</b> 0x290
FCodes.KBD_ALIGN_MIDDLE	KBD_ALIGN_MIDDLE <b>Value</b> 0x291
FCodes.KBD_ALIGN_BOTTOM	KBD_ALIGN_BOTTOM <b>Value</b> 0x292
FCodes.KBD_OBALIGN_TOP	KBD_OBALIGN_TOP <b>Value</b> 0x129

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_OBALIGN_MIDDLE	KBD_OBALIGN_MIDDLE <b>Value</b> 0x12A
FCodes.KBD_OBALIGN_BOTTOM	KBD_OBALIGN_BOTTOM <b>Value</b> 0x12B
FCodes.KBD_OBALIGN_CENTER	KBD_OBALIGN_CENTER <b>Value</b> 0x12C
FCodes.KBD_OBALIGN_LEFT	KBD_OBALIGN_LEFT <b>Value</b> 0x12D
FCodes.KBD_OBALIGN_RIGHT	KBD_OBALIGN_RIGHT <b>Value</b> 0x12E
FCodes.PGF_SINGLE_SPACE	PGF_SINGLE_SPACE <b>Value</b> 0x293
FCodes.PGF_ONEANDAHALF_SPACE	PGF_ONEANDAHALF_SPACE <b>Value</b> 0x294
FCodes.PGF_DOUBLE_SPACE	PGF_DOUBLE_SPACE <b>Value</b> 0x295
FCodes.PGF_SPACE_BETWEEN	PGF_SPACE_BETWEEN <b>Value</b> 0x296
FCodes.PGF_LINE_SPACE	PGF_LINE_SPACE <b>Value</b> 0x297
FCodes.PGF_UPDATE_ALL	PGF_UPDATE_ALL <b>Value</b> 0x298
FCodes.PGF_NEW_FORMAT	PGF_NEW_FORMAT <b>Value</b> 0x299
FCodes.KBD_HELP_INDEX	KBD_HELP_INDEX <b>Value</b> 0x2A0
FCodes.KBD_HELP_KEYS	KBD_HELP_KEYS <b>Value</b> 0x2A1
FCodes.KBD_HELP_SAMPLES	KBD_HELP_SAMPLES <b>Value</b> 0x2A2
FCodes.KBD_HELP_OVERVIEW	KBD_HELP_OVERVIEW <b>Value</b> 0x2A3
FCodes.KBD_HELP_ONLINEMANUALS	KBD_HELP_ONLINEMANUALS <b>Value</b> 0x2A4
FCodes.KBD_HELP_WEBWORKS	KBD_HELP_WEBWORKS <b>Value</b> 0x2A6

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_CUSTOMNEW	KBD_CUSTOMNEW <b>Value 0x2FF</b>
FCodes.KBD_NEW	KBD_NEW <b>Value 0x300</b>
FCodes.KBD_NEWBOOK	KBD_NEWBOOK <b>Value 0x308</b>
FCodes.KBD_OPEN	KBD_OPEN <b>Value 0x301</b>
FCodes.KBD_HEROIC_OPEN	KBD_HEROIC_OPEN <b>Value 0x306</b>
FCodes.KBD_SILENT_OPEN	KBD_SILENT_OPEN <b>Value 0x307</b>
FCodes.KBD_HELP	KBD_HELP <b>Value 0x302</b>
FCodes.KBD_INFO	KBD_INFO <b>Value 0x303</b>
FCodes.KBD_CSHELPMODE	KBD_CSHELPMODE <b>Value 0x304</b>
FCodes.KBD_ABOUTPRODUCT	KBD_ABOUTPRODUCT <b>Value 0x305</b>
FCodes.KBD_BOOKADDFILE	KBD_BOOKADDFILE <b>Value 0x30A</b>
FCodes.KBD_BOOKEDITDEFINE	KBD_BOOKEDITDEFINE <b>Value 0x30B</b>
FCodes.KBD_BOOKRENAMEFILE	KBD_BOOKRENAMEFILE <b>Value 0x30C</b>
FCodes.KBD_BOOKDISPLAYFILENAME	KBD_BOOKDISPLAYFILENAME <b>Value 0x30D</b>
FCodes.KBD_BOOKDISPLAYTEXT	KBD_BOOKDISPLAYTEXT <b>Value 0x30E</b>
FCodes.KBD_BOOKCOMP_EXCLUDE	KBD_BOOKCOMP_EXCLUDE <b>Value 0x4E0</b>
FCodes.KBD_BOOKADDFOLDER	KBD_BOOKADDFOLDER <b>Value 0x601</b>
FCodes.KBD_BOOKCOMP_FILENAME	KBD_BOOKCOMP_FILENAME <b>Value 0x607</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_BOOKCOMP_TEXT	KBD_BOOKCOMP_TEXT <b>Value 0x608</b>
FCodes.KBD_BOOKADDGROUP	KBD_BOOKADDGROUP <b>Value 0x609</b>
FCodes.KBD_MOVE_UP	KBD_MOVE_UP <b>Value 0x70A</b>
FCodes.KBD_MOVE_DOWN	KBD_MOVE_DOWN <b>Value 0x70B</b>
FCodes.KBD_MOVE_LEFT	KBD_MOVE_LEFT <b>Value 0x70C</b>
FCodes.KBD_MOVE_RIGHT	KBD_MOVE_RIGHT <b>Value 0x70D</b>
FCodes.KBD_OPEN_COMPONENTS	KBD_OPEN_COMPONENTS <b>Value 0x3EA</b>
FCodes.KBD_CLOSE_COMPONENTS	KBD_CLOSE_COMPONENTS <b>Value 0x3EB</b>
FCodes.KBD_PRINT_COMPONENTS	KBD_PRINT_COMPONENTS <b>Value 0x3EC</b>
FCodes.KBD_SAVE_COMPONENTS	KBD_SAVE_COMPONENTS <b>Value 0x3ED</b>
FCodes.KBD_COMPONENT_PROPERTIES	KBD_COMPONENT_PROPERTIES <b>Value 0x3EE</b>
FCodes.KBD_DELETE_FILE	KBD_DELETE_FILE <b>Value 0x30F</b>
FCodes.KBD_COLLAPSE_ALL	KBD_COLLAPSE_ALL <b>Value 0x3FA</b>
FCodes.KBD_EXPAND_ALL	KBD_EXPAND_ALL <b>Value 0x3FB</b>
FCodes.SWITCH_TO_RESOURCEMANAGER	SWITCH_TO_RESOURCEMANAGER <b>Value 0x9EA</b>
FCodes.SWITCH_TO_DOCUMENTVIEW	SWITCH_TO_DOCUMENTVIEW <b>Value 0x9EB</b>
FCodes.KBD_SAVE	KBD_SAVE <b>Value 0x310</b>
FCodes.KBD_SAVEAS	KBD_SAVEAS <b>Value 0x311</b>



<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_SAVEASPDFREVIEW	KBD_SAVEASPDFREVIEW <b>Value 0x95A</b>
FCodes.KBD_SAVEASPDFREVIEW2	KBD_SAVEASPDFREVIEW2 <b>Value 0x95B</b>
FCodes.KBD_SAVEASPDFUBIQ	KBD_SAVEASPDFUBIQ <b>Value 0x95E</b>
FCodes.KBD_SAVEASPDFSHARE	KBD_SAVEASPDFSHARE <b>Value 0x95F</b>
FCodes.KBD_SAVEASPDF	KBD_SAVEASPDF <b>Value 0x950</b>
FCodes.KBD_SAVEASXML	KBD_SAVEASXML <b>Value 0x951</b>
FCodes.KBD_MANCOND	KBD_MANCOND <b>Value 0x952</b>
FCodes.KBD_CONDINDICATOR	KBD_CONDINDICATOR <b>Value 0x953</b>
FCodes.KBD_ATTRCOND	KBD_ATTRCOND <b>Value 0x954</b>
FCodes.KBD_PREVIEWFBA	KBD_PREVIEWFBA <b>Value 0xB16</b>
FCodes.AMT_REGISRATION	AMT_REGISRATION <b>Value 0x965</b>
FCodes.AMT_DEACTIVATION	AMT_DEACTIVATION <b>Value 0x967</b>
FCodes.AMT_UPDATES	AMT_UPDATES <b>Value 0x968</b>
FCodes.KBD_REVERT	KBD_REVERT <b>Value 0x312</b>
FCodes.KBD_DOCINFO	KBD_DOCINFO <b>Value 0x3DA</b>
FCodes.KBD_SETPOSTER	KBD_SETPOSTER <b>Value 0x3DB</b>
FCodes.KBD_REMOVEPOSTER	KBD_REMOVEPOSTER <b>Value 0x3DE</b>
FCodes.KBD_PRINT	KBD_PRINT <b>Value 0x313</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_IMPORT	KBD_IMPORT <b>Value 0x314</b>
FCodes.KBD_GENERATE	KBD_GENERATE <b>Value 0x3E1</b>
FCodes.KBD_USEFMFROM	KBD_USEFMFROM <b>Value 0x316</b>
FCodes.KBD_KBMACRO	KBD_KBMACRO <b>Value 0x317</b>
FCodes.KBD_SESSION	KBD_SESSION <b>Value 0x318</b>
FCodes.KBD_PAGESETUP	KBD_PAGESETUP <b>Value 0x319</b>
FCodes.KBD_OPENALL	KBD_OPENALL <b>Value 0x31A</b>
FCodes.KBD_QUITALL	KBD_QUITALL <b>Value 0x31B</b>
FCodes.KBD_SAVEALL	KBD_SAVEALL <b>Value 0x31C</b>
FCodes.KBD_REPEATNEW	KBD_REPEATNEW <b>Value 0x31D</b>
FCodes.KBD_PODLOCATION	KBD_PODLOCATION <b>Value 0x31E</b>
FCodes.KBD_COMPARE	KBD_COMPARE <b>Value 0x31F</b>
FCodes.KBD_UNDO	KBD_UNDO <b>Value 0x320</b>
FCodes.KBD_REDO	KBD_REDO <b>Value 0x935</b>
FCodes.KBD_HIST	KBD_HIST <b>Value 0x936</b>
FCodes.KBD_REPEAT	KBD_REPEAT <b>Value 0x939</b>
FCodes.KBD_CUT	KBD_CUT <b>Value 0x321</b>
FCodes.KBD_COPY	KBD_COPY <b>Value 0x322</b>

Property name	Data Type
FCodes.KBD_PASTE	KBD_PASTE <b>Value</b> 0x323
FCodes.KBD_CLEAR	KBD_CLEAR <b>Value</b> 0x324
FCodes.KBD_COPYFONT	KBD_COPYFONT <b>Value</b> 0x325
FCodes.KBD_COPYPGF	KBD_COPYPGF <b>Value</b> 0x326
FCodes.KBD_SELECTALL	KBD_SELECTALL <b>Value</b> 0x327
FCodes.KBD_STUFF	KBD_STUFF <b>Value</b> 0x328
FCodes.KBD_SEARCH	KBD_SEARCH <b>Value</b> 0x329
FCodes.KBD_SPELLING	KBD_SPELLING <b>Value</b> 0x32A
FCodes.KBD_CAPITAL	KBD_CAPITAL <b>Value</b> 0x32B
FCodes.KBD_YANK	KBD_YANK <b>Value</b> 0x32C
FCodes.KBD_SELECT_GENERATED_FILES	KBD_SELECT_GENERATED_FILES <b>Value</b> 0x32D
FCodes.KBD_SELECT_NON_GENERATED_FILES	KBD_SELECT_NON_GENERATED_FILES <b>Value</b> 0x32E
FCodes.KBD_SELECT_FM_FILES	KBD_SELECT_FM_FILES <b>Value</b> 0x315
FCodes.KBD_SELECT_PRINTABLE_FILES	KBD_SELECT_PRINTABLE_FILES <b>Value</b> 0x9E2
FCodes.KBD_SELECT_EXCLUDED_FILE	KBD_SELECT_EXCLUDED_FILE <b>Value</b> 0x9D1
FCodes.KBD_SELECT_NONEXCLUDED_FILE	KBD_SELECT_NONEXCLUDED_FILE <b>Value</b> 0x9D2
FCodes.KBD_SELECT_CHAPTER_COMPONENTS	KBD_SELECT_CHAPTER_COMPONENTS <b>Value</b> 0x9D3
FCodes.KBD_SELECT_SECTION_COMPONENTS	KBD_SELECT_SECTION_COMPONENTS <b>Value</b> 0x9D4

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_SELECT_SUBSECTION_COMPONENTS	KBD_SELECT_SUBSECTION_COMPONENTS <b>Value 0x9D5</b>
FCodes.KBD_ALLCAP	KBD_ALLCAP <b>Value 0x33A</b>
FCodes.KBD_ALLOWER	KBD_ALLOWER <b>Value 0x33B</b>
FCodes.KBD_INITCAP	KBD_INITCAP <b>Value 0x33C</b>
FCodes.KBD_ALLCAPH	KBD_ALLCAPH <b>Value 0x369</b>
FCodes.KBD_ALLOWERH	KBD_ALLOWERH <b>Value 0x36A</b>
FCodes.KBD_INITCAPH	KBD_INITCAPH <b>Value 0x36B</b>
FCodes.KBD_THESAURUS	KBD_THESAURUS <b>Value 0x3D0</b>
FCodes.KBD_THESAURUS_REPLACE	KBD_THESAURUS_REPLACE <b>Value 0x3D3</b>
FCodes.KBD_CREATE_PUBLISHER	KBD_CREATE_PUBLISHER <b>Value 0x3D5</b>
FCodes.KBD_SUBSCRIBE_TO	KBD_SUBSCRIBE_TO <b>Value 0x3D6</b>
FCodes.KBD_LINK_BOUNDARIES	KBD_LINK_BOUNDARIES <b>Value 0x3D7</b>
FCodes.KBD_FONTDESIGN	KBD_FONTDESIGN <b>Value 0x330</b>
FCodes.KBD_PGFDESIGN	KBD_PGFDESIGN <b>Value 0x331</b>
FCodes.KBD_RUBIPROPS	KBD_RUBIPROPS <b>Value 0x3DD</b>
FCodes.KBD_COLLAYOUT	KBD_COLLAYOUT <b>Value 0x348</b>
FCodes.KBD_LINELAYOUT	KBD_LINELAYOUT <b>Value 0x332</b>
FCodes.KBD_PAGESIZE	KBD_PAGESIZE <b>Value 0x349</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_PAGINATION	KBD_PAGINATION <b>Value 0xA40</b>
FCodes.KBD_PAGEBACK	KBD_PAGEBACK <b>Value 0x34A</b>
FCodes.KBD_NEWMMASTER	KBD_NEWMMASTER <b>Value 0xA0A</b>
FCodes.KBD_REORDERMASTER	KBD_REORDERMASTER <b>Value 0xA0B</b>
FCodes.KBD_PAGEUPDATE	KBD_PAGEUPDATE <b>Value 0xA00</b>
FCodes.KBD_CUST_TEXT_FRAME	KBD_CUST_TEXT_FRAME <b>Value 0xA01</b>
FCodes.KBD_CONNECT_TEXT_FRAME	KBD_CONNECT_TEXT_FRAME <b>Value 0x35B</b>
FCodes.KBD_CUTHEAD	KBD_CUTHEAD <b>Value 0x35C</b>
FCodes.KBD_CURTAIL	KBD_CURTAIL <b>Value 0x35D</b>
FCodes.KBD_CUTBOTH	KBD_CUTBOTH <b>Value 0xA04</b>
FCodes.KBD_SPLIT	KBD_SPLIT <b>Value 0x35A</b>
FCodes.KBD_SPLITR	KBD_SPLITR <b>Value 0xA02</b>
FCodes.KBD_SPLITL	KBD_SPLITL <b>Value 0xA03</b>
FCodes.KBD_ROTPAGE_PLUS	KBD_ROTPAGE_PLUS <b>Value 0x34E</b>
FCodes.KBD_ROTPAGE_MINUS	KBD_ROTPAGE_MINUS <b>Value 0x34F</b>
FCodes.KBD_ROTPAGE_NORM	KBD_ROTPAGE_NORM <b>Value 0xA05</b>
FCodes.KBD_NUMBERING	KBD_NUMBERING <b>Value 0x333</b>
FCodes.KBD_CBARPRO	KBD_CBARPRO <b>Value 0x334</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_FOOTNOTEPRO	KBD_FOOTNOTEPRO <b>Value 0x335</b>
FCodes.KBD_TEXT_OPTIONS	KBD_TEXT_OPTIONS <b>Value 0x337</b>
FCodes.KBD_COMBINED_FONTS	KBD_COMBINED_FONTS <b>Value 0x338</b>
FCodes.KBD_ACROBAT_SETUP	KBD_ACROBAT_SETUP <b>Value 0x36C</b>
FCodes.KBD_LINENUMPRO	KBD_LINENUMPRO <b>Value 0x673</b>
FCodes.KBD_LINENUMTOGGLE	KBD_LINENUMTOGGLE <b>Value 0x674</b>
FCodes.KBD_FIRSTPAGE	KBD_FIRSTPAGE <b>Value 0x340</b>
FCodes.KBD_LASTPAGE	KBD_LASTPAGE <b>Value 0x341</b>
FCodes.KBD_BODYPAGE	KBD_BODYPAGE <b>Value 0x342</b>
FCodes.KBD_MASTERPAGE	KBD_MASTERPAGE <b>Value 0x343</b>
FCodes.KBD_REFPAGE	KBD_REFPAGE <b>Value 0x344</b>
FCodes.KBD_GOTOPAGE	KBD_GOTOPAGE <b>Value 0x345</b>
FCodes.KBD_ADDPAGE	KBD_ADDPAGE <b>Value 0x346</b>
FCodes.KBD_DELETEPAGE	KBD_DELETEPAGE <b>Value 0x347</b>
FCodes.KBD_FREEZE	KBD_FREEZE <b>Value 0x34B</b>
FCodes.KBD_TEXTCOLPRO	KBD_TEXTCOLPRO <b>Value 0x339</b>
FCodes.KBD_PREVPAGE	KBD_PREVPAGE <b>Value 0x34C</b>
FCodes.KBD_NEXTPAGE	KBD_NEXTPAGE <b>Value 0x34D</b>

Property name	Data Type
FCodes.KBD_GOTOPAGEN	KBD_GOTOPAGEN <b>Value</b> 0x96E
FCodes.KBD_GOTOIP	KBD_GOTOIP <b>Value</b> 0x96F
FCodes.KBD_STRWIN_LEFTANCHOR	KBD_STRWIN_LEFTANCHOR <b>Value</b> 0x3DF
FCodes.KBD_PAGEBREAK	KBD_PAGEBREAK <b>Value</b> 0x32F
FCodes.KBD_ANCHOR	KBD_ANCHOR <b>Value</b> 0x350
FCodes.KBD_FOOTNOTE	KBD_FOOTNOTE <b>Value</b> 0x351
FCodes.KBD_REFERENCE	KBD_REFERENCE <b>Value</b> 0x352
FCodes.KBD_VARIABLE	KBD_VARIABLE <b>Value</b> 0x353
FCodes.KBD_INSET	KBD_INSET <b>Value</b> 0x354
FCodes.KBD_HYPERTEXT	KBD_HYPERTEXT <b>Value</b> 0x359
FCodes.KBD_MARKERS	KBD_MARKERS <b>Value</b> 0x355
FCodes.KBD_EQUATION	KBD_EQUATION <b>Value</b> 0x336
FCodes.KBD_CONDTEXT	KBD_CONDTEXT <b>Value</b> 0x357
FCodes.KBD_NEWMARKER	KBD_NEWMARKER <b>Value</b> 0x356
FCodes.KBD_EDITMARKERTYPE	KBD_EDITMARKERTYPE <b>Value</b> 0x409
FCodes.KBD_DELMARKERTYPE	KBD_DELMARKERTYPE <b>Value</b> 0x358
FCodes.KBD_RENAMEMARKERTYPE	KBD_RENAMEMARKERTYPE <b>Value</b> 0x35F
FCodes.KBD_ADDMARKERTYPE	KBD_ADDMARKERTYPE <b>Value</b> 0x41D

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_DELETEMARKER	KBD_DELETEMARKER <b>Value 0x969</b>
FCodes.KBD_MARKERSPOD	KBD_MARKERSPOD <b>Value 0x96A</b>
FCodes.KBD_XREFSPOD	KBD_XREFSPOD <b>Value 0x96B</b>
FCodes.KBD_FONTPOD	KBD_FONTPOD <b>Value 0x966</b>
FCodes.KBD_FONTREPLACEPOD	KBD_FONTREPLACEPOD <b>Value 0x95C</b>
FCodes.KBD_HOTSPOTSPOD	KBD_HOTSPOTSPOD <b>Value 0x4FF</b>
FCodes.KBD_NEWAFRAME	KBD_NEWAFRAME <b>Value 0x96D</b>
FCodes.KBD_NEWHYPERTEXT	KBD_NEWHYPERTEXT <b>Value 0x35E</b>
FCodes.KBD_VALIDATE_HYPERTEXT	KBD_VALIDATE_HYPERTEXT <b>Value 0xF13</b>
FCodes.MENU_HYPERTEXT	MENU_HYPERTEXT <b>Value 0xC18</b>
FCodes.KBD_HYPRTXT_SHTCUT	KBD_HYPRTXT_SHTCUT <b>Value 0x41E</b>
FCodes.WEB_GOTOADOBE	WEB_GOTOADOBE <b>Value 0xC19</b>
FCodes.KBD_FMPIP	KBD_FMPIP <b>Value 0x806</b>
FCodes.WEB_PREFERENCES	WEB_PREFERENCES <b>Value 0xC1A</b>
FCodes.WEB_TOPISSUES	WEB_TOPISSUES <b>Value 0xC1B</b>
FCodes.WEB_ADOBEHELP	WEB_ADOBEHELP <b>Value 0xC22</b>
FCodes.WEB_CORPORATENEWS	WEB_CORPORATENEWS <b>Value 0xC1D</b>
FCodes.WEB_REGISTRATION	WEB_REGISTRATION <b>Value 0xC1E</b>



<b>Property name</b>	<b>Data Type</b>
FCodes.WEB_FRAME_BOOKMARKS	WEB_FRAME_BOOKMARKS <b>Value 0xC1F</b>
FCodes.WEB_ADOBE_BOOKMARKS	WEB_ADOBE_BOOKMARKS <b>Value 0xC21</b>
FCodes.KBD_DUMPHYPERTEXT	KBD_DUMPHYPERTEXT <b>Value 0xC20</b>
FCodes.KBD_VARCURPG	KBD_VARCURPG <b>Value 0xA06</b>
FCodes.KBD_VARPGCOUNT	KBD_VARPGCOUNT <b>Value 0xA07</b>
FCodes.KBD_VARCURDATE	KBD_VARCURDATE <b>Value 0xA08</b>
FCodes.KBD_VAROTHER	KBD_VAROTHER <b>Value 0xA09</b>
FCodes.KBD_EDITVARIABLE	KBD_EDITVARIABLE <b>Value 0xA0C</b>
FCodes.KBD_FINDNEXT	KBD_FINDNEXT <b>Value 0x955</b>
FCodes.KBD_FINDPREV	KBD_FINDPREV <b>Value 0x956</b>
FCodes.KBD_TRACKCHANGE	KBD_TRACKCHANGE <b>Value 0x957</b>
FCodes.KBD_TRACKCHANGEDISABLE	KBD_TRACKCHANGEDISABLE <b>Value 0x95D</b>
FCodes.KBD_ACCEPTCHANGE	KBD_ACCEPTCHANGE <b>Value 0x958</b>
FCodes.KBD_REJECTCHANGE	KBD_REJECTCHANGE <b>Value 0x959</b>
FCodes.KBD_ACCEPTCHANGEANDNEXT	KBD_ACCEPTCHANGEANDNEXT <b>Value 0x99B</b>
FCodes.KBD_REJECTCHANGEANDNEXT	KBD_REJECTCHANGEANDNEXT <b>Value 0x99C</b>
FCodes.KBD_ACCEPTALLCHANGE	KBD_ACCEPTALLCHANGE <b>Value 0x960</b>
FCodes.KBD_REJECTALLCHANGE	KBD_REJECTALLCHANGE <b>Value 0x961</b>

## Object Reference

Property name	Data Type
FCodes.KBD_PREVIEW_ACCEPTALL	KBD_PREVIEW_ACCEPTALL <b>Value</b> 0x962
FCodes.KBD_PREVIEW_REJECTALL	KBD_PREVIEW_REJECTALL <b>Value</b> 0x963
FCodes.KBD_PREVIEW_OFF	KBD_PREVIEW_OFF <b>Value</b> 0x964
FCodes.KBD_CONFIGURE_TC_COLOR	KBD_CONFIGURE_TC_COLOR <b>Value</b> 0x990
FCodes.KBD_TC_USER_NAME	KBD_TC_USER_NAME <b>Value</b> 0x991
FCodes.KBD_TC_SEARCH_DOC	KBD_TC_SEARCH_DOC <b>Value</b> 0x133
FCodes.KBD_TC_SEARCH_BOOK	KBD_TC_SEARCH_BOOK <b>Value</b> 0x134
FCodes.KBD_TC_SEARCH_DITAMAP	KBD_TC_SEARCH_DITAMAP <b>Value</b> 0x135
FCodes.KBD_TC_SEARCH_SEL	KBD_TC_SEARCH_SEL <b>Value</b> 0x136
FCodes.KBD_TOC	KBD_TOC <b>Value</b> 0x2B0
FCodes.KBD_LIST_FIGURE	KBD_LIST_FIGURE <b>Value</b> 0x2B1
FCodes.KBD_LIST_TABLE	KBD_LIST_TABLE <b>Value</b> 0x2B2
FCodes.KBD_LIST_PGF	KBD_LIST_PGF <b>Value</b> 0x2B3
FCodes.KBD_LIST_PGF_ALPHA	KBD_LIST_PGF_ALPHA <b>Value</b> 0x2B4
FCodes.KBD_LIST_MARKER	KBD_LIST_MARKER <b>Value</b> 0x2B5
FCodes.KBD_LIST_MARKER_ALPHA	KBD_LIST_MARKER_ALPHA <b>Value</b> 0x2B6
FCodes.KBD_LIST_REFERENCES	KBD_LIST_REFERENCES <b>Value</b> 0x2B7
FCodes.KBD_INDEX_STANDARD	KBD_INDEX_STANDARD <b>Value</b> 0x2C0

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_INDEX_AUTHOR	KBD_INDEX_AUTHOR <b>Value 0x2C1</b>
FCodes.KBD_INDEX_SUBJECT	KBD_INDEX_SUBJECT <b>Value 0x2C2</b>
FCodes.KBD_INDEX_MARKER	KBD_INDEX_MARKER <b>Value 0x2C3</b>
FCodes.KBD_INDEX_REFERENCES	KBD_INDEX_REFERENCES <b>Value 0x2C4</b>
FCodes.KBD_OPTIONS	KBD_OPTIONS <b>Value 0x360</b>
FCodes.KBD_BORDERS	KBD_BORDERS <b>Value 0x361</b>
FCodes.KBD_SYMBOLS	KBD_SYMBOLS <b>Value 0x362</b>
FCodes.KBD_RULERS	KBD_RULERS <b>Value 0x363</b>
FCodes.KBD_GRID	KBD_GRID <b>Value 0x364</b>
FCodes.KBD_SEPARATIONS	KBD_SEPARATIONS <b>Value 0x365</b>
FCodes.KBD_COLOR	KBD_COLOR <b>Value 0x39F</b>
FCodes.KBD_CONDVISIBILITY	KBD_CONDVISIBILITY <b>Value 0x367</b>
FCodes.KBD_CONDTOGGLEOVERR	KBD_CONDTOGGLEOVERR <b>Value 0x368</b>
FCodes.KBD_HOTSPOTINDICATORS	KBD_HOTSPOTINDICATORS <b>Value 0x508</b>
FCodes.KBD_TOGGLEDRAW	KBD_TOGGLEDRAW <b>Value 0x366</b>
FCodes.KBD_VIEWSEP1	KBD_VIEWSEP1 <b>Value 0x36D</b>
FCodes.KBD_VIEWSEP2	KBD_VIEWSEP2 <b>Value 0x36E</b>
FCodes.KBD_VIEWSEP3	KBD_VIEWSEP3 <b>Value 0x36F</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_VIEWSEP4	KBD_VIEWSEP4 <b>Value</b> 0x37D
FCodes.KBD_VIEWSEP5	KBD_VIEWSEP5 <b>Value</b> 0x37E
FCodes.KBD_VIEWSEP6	KBD_VIEWSEP6 <b>Value</b> 0x39E
FCodes.KBD_MENUCOMPLETE	KBD_MENUCOMPLETE <b>Value</b> 0x33D
FCodes.KBD_MENUQUICK	KBD_MENUQUICK <b>Value</b> 0x33E
FCodes.KBD_MENUCUSTOM	KBD_MENUCUSTOM <b>Value</b> 0x33F
FCodes.KBD_MENUMODIFY	KBD_MENUMODIFY <b>Value</b> 0x3D9
FCodes.KBD_MENURESET	KBD_MENURESET <b>Value</b> 0x3D8
FCodes.KBD_SHOW_BORDERS	KBD_SHOW_BORDERS <b>Value</b> 0x3F1
FCodes.KBD_SHOW_HOTSPOTINDICATORS	KBD_SHOW_HOTSPOTINDICATORS <b>Value</b> 0x509
FCodes.KBD_SHOW_SYMBOLS	KBD_SHOW_SYMBOLS <b>Value</b> 0x3F2
FCodes.KBD_SHOW_RULERS	KBD_SHOW_RULERS <b>Value</b> 0x3F3
FCodes.KBD_SHOW_GRID	KBD_SHOW_GRID <b>Value</b> 0x3F4
FCodes.KBD_SHOW_LINK_BOUNDARIES	KBD_SHOW_LINK_BOUNDARIES <b>Value</b> 0x3F5
FCodes.KBD_SHOW_ELEM_BORDER	KBD_SHOW_ELEM_BORDER <b>Value</b> 0x3F6
FCodes.KBD_SHOW_ELEM_TAGS	KBD_SHOW_ELEM_TAGS <b>Value</b> 0x3F7
FCodes.KBD_SHOW_GRAPHICS	KBD_SHOW_GRAPHICS <b>Value</b> 0x3F8
FCodes.KBD_SHOW_COND_IND	KBD_SHOW_COND_IND <b>Value</b> 0x3F9

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_HIDE_BORDERS	KBD_HIDE_BORDERS <b>Value 0x4F1</b>
FCodes.KBD_HIDE_HOTSPOTINDICATORS	KBD_HIDE_HOTSPOTINDICATORS <b>Value 0x50A</b>
FCodes.KBD_HIDE_SYMBOLS	KBD_HIDE_SYMBOLS <b>Value 0x4F2</b>
FCodes.KBD_HIDE_RULERS	KBD_HIDE_RULERS <b>Value 0x4F3</b>
FCodes.KBD_HIDE_GRID	KBD_HIDE_GRID <b>Value 0x4F4</b>
FCodes.KBD_HIDE_LINK_BOUNDARIES	KBD_HIDE_LINK_BOUNDARIES <b>Value 0x4F5</b>
FCodes.KBD_HIDE_ELEM_BORDER	KBD_HIDE_ELEM_BORDER <b>Value 0x4F6</b>
FCodes.KBD_HIDE_GRAPHICS	KBD_HIDE_GRAPHICS <b>Value 0x4F8</b>
FCodes.KBD_HIDE_COND_IND	KBD_HIDE_COND_IND <b>Value 0x4F9</b>
FCodes.KBD_INSETPOD	KBD_INSETPOD <b>Value 0x994</b>
FCodes.KBD_SEARCHREFS	KBD_SEARCHREFS <b>Value 0xB17</b>
FCodes.KBD_FLIPUD	KBD_FLIPUD <b>Value 0x370</b>
FCodes.KBD_FLIPLR	KBD_FLIPLR <b>Value 0x371</b>
FCodes.KBD_ROTATE_CCW	KBD_ROTATE_CCW <b>Value 0x372</b>
FCodes.KBD_ROT_PLUS	KBD_ROT_PLUS <b>Value 0x372</b>
FCodes.KBD_ROTATE_CCW_SMALL	KBD_ROTATE_CCW_SMALL <b>Value 0x38A</b>
FCodes.KBD_SCALE	KBD_SCALE <b>Value 0x373</b>
FCodes.KBD_SMOOTH	KBD_SMOOTH <b>Value 0x374</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_UNSMOOTH	KBD_UNSMOOTH <b>Value 0x375</b>
FCodes.KBD_RESHAPE	KBD_RESHAPE <b>Value 0x376</b>
FCodes.KBD_JOINCURVES	KBD_JOINCURVES <b>Value 0x377</b>
FCodes.KBD_SETSIDES	KBD_SETSIDES <b>Value 0x378</b>
FCodes.KBD_CONSTRAIN	KBD_CONSTRAIN <b>Value 0x379</b>
FCodes.KBD_SNAP	KBD_SNAP <b>Value 0x37A</b>
FCodes.KBD_GRAVITY	KBD_GRAVITY <b>Value 0x37B</b>
FCodes.KBD_KEEPTOOL	KBD_KEEPTOOL <b>Value 0x37C</b>
FCodes.KBD_OVERPRINT	KBD_OVERPRINT <b>Value 0x3D2</b>
FCodes.KBD_OVERPRINT_NONE	KBD_OVERPRINT_NONE <b>Value 0x3D4</b>
FCodes.KBD_RUN_OFF	KBD_RUN_OFF <b>Value 0x670</b>
FCodes.KBD_RUN_CONTOUR	KBD_RUN_CONTOUR <b>Value 0x668</b>
FCodes.KBD_RUN_BBOX	KBD_RUN_BBOX <b>Value 0x669</b>
FCodes.KBD_RUN_GAP	KBD_RUN_GAP <b>Value 0x671</b>
FCodes.KBD_RUN_PROPS	KBD_RUN_PROPS <b>Value 0x667</b>
FCodes.KBD_SETRUN_PROPS	KBD_SETRUN_PROPS <b>Value 0x96C</b>
FCodes.KBD_FRONT	KBD_FRONT <b>Value 0x380</b>
FCodes.KBD_BACK	KBD_BACK <b>Value 0x381</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_GROUP	KBD_GROUP <b>Value 0x382</b>
FCodes.KBD_UNGROUP	KBD_UNGROUP <b>Value 0x383</b>
FCodes.KBD_ALIGN	KBD_ALIGN <b>Value 0x384</b>
FCodes.KBD_DISTRIBUTE	KBD_DISTRIBUTE <b>Value 0x385</b>
FCodes.KBD_ROTATE_CW	KBD_ROTATE_CW <b>Value 0x386</b>
FCodes.KBD_ROT_MINUS	KBD_ROT_MINUS <b>Value 0x386</b>
FCodes.KBD_ROTATE_CW_SMALL	KBD_ROTATE_CW_SMALL <b>Value 0x389</b>
FCodes.KBD_OBJPROPS	KBD_OBJPROPS <b>Value 0x387</b>
FCodes.KBD_PICKOBJPROPS	KBD_PICKOBJPROPS <b>Value 0x388</b>
FCodes.KBD_MUTATE	KBD_MUTATE <b>Value 0x37F</b>
FCodes.KBD_ROTATE	KBD_ROTATE <b>Value 0x38B</b>
FCodes.KBD_ROTATE_AGAIN	KBD_ROTATE_AGAIN <b>Value 0x38C</b>
FCodes.KBD_ROTATE_NATURAL	KBD_ROTATE_NATURAL <b>Value 0x38D</b>
FCodes.KBD_REROTATE	KBD_REROTATE <b>Value 0x38E</b>
FCodes.KBD_ROTATE_RESET0	KBD_ROTATE_RESET0 <b>Value 0x38F</b>
FCodes.KBD_SETALIGN_PROPS	KBD_SETALIGN_PROPS <b>Value 0x992</b>
FCodes.KBD_SETDISTRIBUTE_PROPS	KBD_SETDISTRIBUTE_PROPS <b>Value 0x993</b>
FCodes.KBD_CLOSEWIN	KBD_CLOSEWIN <b>Value 0x390</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_OPENWIN	KBD_OPENWIN <b>Value 0x391</b>
FCodes.KBD_CLOPWIN	KBD_CLOPWIN <b>Value 0x392</b>
FCodes.KBD_MOVEWIN	KBD_MOVEWIN <b>Value 0x393</b>
FCodes.KBD_EXPOSEWIN	KBD_EXPOSEWIN <b>Value 0x395</b>
FCodes.KBD_HIDEWIN	KBD_HIDEWIN <b>Value 0x396</b>
FCodes.KBD_HISHWIN	KBD_HISHWIN <b>Value 0x397</b>
FCodes.KBD_REFRESHWIN	KBD_REFRESHWIN <b>Value 0x398</b>
FCodes.KBD_QUITWIN	KBD_QUITWIN <b>Value 0x399</b>
FCodes.KBD_RESIZELOCK	KBD_RESIZELOCK <b>Value 0x39A</b>
FCodes.KBD_RESIZEUNLOCK	KBD_RESIZEUNLOCK <b>Value 0x39B</b>
FCodes.KBD_QUIETCLOSEWIN	KBD_QUIETCLOSEWIN <b>Value 0x39C</b>
FCodes.KBD_POPUP_MENU	KBD_POPUP_MENU <b>Value 0x39D</b>
FCodes.KBD_LINETOOL	KBD_LINETOOL <b>Value 0x3A0</b>
FCodes.KBD_RECTTOOL	KBD_RECTTOOL <b>Value 0x3A1</b>
FCodes.KBD_POLYGTOOL	KBD_POLYGTOOL <b>Value 0x3A2</b>
FCodes.KBD_POLYLTOOL	KBD_POLYLTOOL <b>Value 0x3A3</b>
FCodes.KBD_ARCTOOL	KBD_ARCTOOL <b>Value 0x3A4</b>
FCodes.KBD_ROUNDRECT	KBD_ROUNDRECT <b>Value 0x3A5</b>



<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_OVALTOOL	KBD_OVALTOOL <b>Value 0x3A6</b>
FCodes.KBD_TEXTLTOOL	KBD_TEXTLTOOL <b>Value 0x3A7</b>
FCodes.KBD_TEXTRTOOL	KBD_TEXTRTOOL <b>Value 0x3A8</b>
FCodes.KBD_FREETOOL	KBD_FREETOOL <b>Value 0x3A9</b>
FCodes.KBD_FRAMETOOL	KBD_FRAMETOOL <b>Value 0x3AA</b>
FCodes.KBD_LASTTOOL	KBD_LASTTOOL <b>Value 0x3AB</b>
FCodes.KBD_WIDTH0	KBD_WIDTH0 <b>Value 0x3AC</b>
FCodes.KBD_WIDTH1	KBD_WIDTH1 <b>Value 0x3AD</b>
FCodes.KBD_INCWIDTH	KBD_INCWIDTH <b>Value 0x3AE</b>
FCodes.KBD_DECWIDTH	KBD_DECWIDTH <b>Value 0x3AF</b>
FCodes.KBD_PEN0	KBD_PEN0 <b>Value 0x3B0</b>
FCodes.KBD_PEN1	KBD_PEN1 <b>Value 0x3B1</b>
FCodes.KBD_INCPEN	KBD_INCPEN <b>Value 0x3B2</b>
FCodes.KBD_DECPEN	KBD_DECPEN <b>Value 0x3B3</b>
FCodes.KBD_FILL0	KBD_FILL0 <b>Value 0x3B4</b>
FCodes.KBD_FILL1	KBD_FILL1 <b>Value 0x3B5</b>
FCodes.KBD_INCFILL	KBD_INCFILL <b>Value 0x3B6</b>
FCodes.KBD_DECFILL	KBD_DECFILL <b>Value 0x3B7</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_SETFILL	KBD_SETFILL <b>Value 0x3B8</b>
FCodes.KBD_SETPEN	KBD_SETPEN <b>Value 0x3B9</b>
FCodes.KBD_SETWIDTH	KBD_SETWIDTH <b>Value 0x3BA</b>
FCodes.KBD_SETCAP	KBD_SETCAP <b>Value 0x3BB</b>
FCodes.KBD_SETSEP	KBD_SETSEP <b>Value 0x3BC</b>
FCodes.KBD_SETSEP_KEEP	KBD_SETSEP_KEEP <b>Value 0x3BE</b>
FCodes.KBD_SETSEP_ALL	KBD_SETSEP_ALL <b>Value 0x3BF</b>
FCodes.KBD_SETSEP_RESET_TINT_OVERPRINT	KBD_SETSEP_RESET_TINT_OVERPRINT <b>Value 0x3D1</b>
FCodes.KBD_SETFILL_0	KBD_SETFILL_0 <b>Value 0x430</b>
FCodes.KBD_SETFILL_1	KBD_SETFILL_1 <b>Value 0x431</b>
FCodes.KBD_SETFILL_2	KBD_SETFILL_2 <b>Value 0x432</b>
FCodes.KBD_SETFILL_3	KBD_SETFILL_3 <b>Value 0x433</b>
FCodes.KBD_SETFILL_4	KBD_SETFILL_4 <b>Value 0x434</b>
FCodes.KBD_SETFILL_5	KBD_SETFILL_5 <b>Value 0x435</b>
FCodes.KBD_SETFILL_6	KBD_SETFILL_6 <b>Value 0x436</b>
FCodes.KBD_SETFILL_7	KBD_SETFILL_7 <b>Value 0x437</b>
FCodes.KBD_SETFILL_8	KBD_SETFILL_8 <b>Value 0x438</b>
FCodes.KBD_SETFILL_9	KBD_SETFILL_9 <b>Value 0x439</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_SETFILL_A	KBD_SETFILL_A <b>Value 0x43A</b>
FCodes.KBD_SETFILL_B	KBD_SETFILL_B <b>Value 0x43B</b>
FCodes.KBD_SETFILL_C	KBD_SETFILL_C <b>Value 0x43C</b>
FCodes.KBD_SETFILL_D	KBD_SETFILL_D <b>Value 0x43D</b>
FCodes.KBD_SETFILL_E	KBD_SETFILL_E <b>Value 0x43E</b>
FCodes.KBD_SETFILL_F	KBD_SETFILL_F <b>Value 0x43F</b>
FCodes.KBD_SETPEN_0	KBD_SETPEN_0 <b>Value 0x440</b>
FCodes.KBD_SETPEN_1	KBD_SETPEN_1 <b>Value 0x441</b>
FCodes.KBD_SETPEN_2	KBD_SETPEN_2 <b>Value 0x442</b>
FCodes.KBD_SETPEN_3	KBD_SETPEN_3 <b>Value 0x443</b>
FCodes.KBD_SETPEN_4	KBD_SETPEN_4 <b>Value 0x444</b>
FCodes.KBD_SETPEN_5	KBD_SETPEN_5 <b>Value 0x445</b>
FCodes.KBD_SETPEN_6	KBD_SETPEN_6 <b>Value 0x446</b>
FCodes.KBD_SETPEN_7	KBD_SETPEN_7 <b>Value 0x447</b>
FCodes.KBD_SETPEN_8	KBD_SETPEN_8 <b>Value 0x448</b>
FCodes.KBD_SETPEN_9	KBD_SETPEN_9 <b>Value 0x449</b>
FCodes.KBD_SETPEN_A	KBD_SETPEN_A <b>Value 0x44A</b>
FCodes.KBD_SETPEN_B	KBD_SETPEN_B <b>Value 0x44B</b>

## Object Reference

Property name	Data Type
FCodes.KBD_SETPEN_C	KBD_SETPEN_C <b>Value</b> 0x44C
FCodes.KBD_SETPEN_D	KBD_SETPEN_D <b>Value</b> 0x44D
FCodes.KBD_SETPEN_E	KBD_SETPEN_E <b>Value</b> 0x44E
FCodes.KBD_SETPEN_F	KBD_SETPEN_F <b>Value</b> 0x44F
FCodes.KBD_SETWIDTH_0	KBD_SETWIDTH_0 <b>Value</b> 0x450
FCodes.KBD_SETWIDTH_1	KBD_SETWIDTH_1 <b>Value</b> 0x451
FCodes.KBD_SETWIDTH_2	KBD_SETWIDTH_2 <b>Value</b> 0x452
FCodes.KBD_SETWIDTH_3	KBD_SETWIDTH_3 <b>Value</b> 0x453
FCodes.KBD_SETWIDTH_SLIDE	KBD_SETWIDTH_SLIDE <b>Value</b> 0x45E
FCodes.KBD_SETWIDTH_OPTION	KBD_SETWIDTH_OPTION <b>Value</b> 0x45F
FCodes.KBD_SETCAP_0	KBD_SETCAP_0 <b>Value</b> 0x460
FCodes.KBD_SETCAP_1	KBD_SETCAP_1 <b>Value</b> 0x461
FCodes.KBD_SETCAP_2	KBD_SETCAP_2 <b>Value</b> 0x462
FCodes.KBD_SETCAP_3	KBD_SETCAP_3 <b>Value</b> 0x463
FCodes.KBD_SETCAP_OPTION	KBD_SETCAP_OPTION <b>Value</b> 0x46F
FCodes.KBD_SETDASH_0	KBD_SETDASH_0 <b>Value</b> 0x470
FCodes.KBD_SETDASH_1	KBD_SETDASH_1 <b>Value</b> 0x471
FCodes.KBD_SETDASH_2	KBD_SETDASH_2 <b>Value</b> 0x472

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_SETDASH_3	KBD_SETDASH_3 <b>Value 0x473</b>
FCodes.KBD_SETDASH_4	KBD_SETDASH_4 <b>Value 0x474</b>
FCodes.KBD_SETDASH_5	KBD_SETDASH_5 <b>Value 0x475</b>
FCodes.KBD_SETDASH_6	KBD_SETDASH_6 <b>Value 0x476</b>
FCodes.KBD_SETDASH_7	KBD_SETDASH_7 <b>Value 0x477</b>
FCodes.KBD_SETDASH_8	KBD_SETDASH_8 <b>Value 0x478</b>
FCodes.KBD_SETDASH_OPTION	KBD_SETDASH_OPTION <b>Value 0x47F</b>
FCodes.KBD_SNAP_0	KBD_SNAP_0 <b>Value 0x480</b>
FCodes.KBD_SNAP_1	KBD_SNAP_1 <b>Value 0x481</b>
FCodes.KBD_GRAVITY_0	KBD_GRAVITY_0 <b>Value 0x482</b>
FCodes.KBD_GRAVITY_1	KBD_GRAVITY_1 <b>Value 0x483</b>
FCodes.KBD_SETKNOCKOUT	KBD_SETKNOCKOUT <b>Value 0x48A</b>
FCodes.KBD_SETOVERPRINT	KBD_SETOVERPRINT <b>Value 0x48B</b>
FCodes.KBD_SETFROMCOLOR	KBD_SETFROMCOLOR <b>Value 0x48C</b>
FCodes.KBD_SETTINT	KBD_SETTINT <b>Value 0x48D</b>
FCodes.KBD_SETSOLID	KBD_SETSOLID <b>Value 0x402</b>
FCodes.KBD_SETDASH	KBD_SETDASH <b>Value 0x403</b>
FCodes.KBD_DASH0	KBD_DASH0 <b>Value 0x404</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_DASH1	KBD_DASH1 <b>Value 0x405</b>
FCodes.KBD_INCDASH	KBD_INCDASH <b>Value 0x406</b>
FCodes.KBD_DECDASH	KBD_DECDASH <b>Value 0x407</b>
FCodes.KBD_DASHOPTION	KBD_DASHOPTION <b>Value 0x408</b>
FCodes.KBD_CHECKSEL	KBD_CHECKSEL <b>Value 0x3C0</b>
FCodes.KBD_CHECKDOC	KBD_CHECKDOC <b>Value 0x3C1</b>
FCodes.KBD_CORRECT	KBD_CORRECT <b>Value 0x3C2</b>
FCodes.KBD_ADDUSRDICT	KBD_ADDUSRDICT <b>Value 0x3C3</b>
FCodes.KBD_ADDDOCDICT	KBD_ADDDOCDICT <b>Value 0x3C4</b>
FCodes.KBD_ADDAUTOCORR	KBD_ADDAUTOCORR <b>Value 0x3C5</b>
FCodes.KBD_DELUSRDICT	KBD_DELUSRDICT <b>Value 0x3C6</b>
FCodes.KBD_DELDOCDICT	KBD_DELDOCDICT <b>Value 0x3C7</b>
FCodes.KBD_CLEARAUTO	KBD_CLEARAUTO <b>Value 0x3C8</b>
FCodes.KBD_CHANGEDICT	KBD_CHANGEDICT <b>Value 0x3C9</b>
FCodes.KBD_SPELLRESET	KBD_SPELLRESET <b>Value 0x3CA</b>
FCodes.KBD_CHECKPAGE	KBD_CHECKPAGE <b>Value 0x3CB</b>
FCodes.KBD_SPOPTIONS	KBD_SPOPTIONS <b>Value 0x3CC</b>
FCodes.KBD_HYPHENATE	KBD_HYPHENATE <b>Value 0x3CD</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_CHECKBATCH	KBD_CHECKBATCH <b>Value 0x3CE</b>
FCodes.KBD_REFORMATDOC	KBD_REFORMATDOC <b>Value 0x3CF</b>
FCodes.KBD_ASC_CORRECT	KBD_ASC_CORRECT <b>Value 0x995</b>
FCodes.KBD_ASC_ADDUSRDICT	KBD_ASC_ADDUSRDICT <b>Value 0x996</b>
FCodes.KBD_ASC_ADDDOCDICT	KBD_ASC_ADDDOCDICT <b>Value 0x997</b>
FCodes.KBD_ASC_ENABLE_AUTO_SPELL_CHECK	KBD_ASC_ENABLE_AUTO_SPELL_CHECK <b>Value 0x999</b>
FCodes.TEXT_INSET_PROPS	TEXT_INSET_PROPS <b>Value 0x3E0</b>
FCodes.UPDATE_INSETS	UPDATE_INSETS <b>Value 0x3E1</b>
FCodes.SUPPRESS_INSETS	SUPPRESS_INSETS <b>Value 0x3E2</b>
FCodes.KBD_CREATE_MULTIMEDIA_LINK	KBD_CREATE_MULTIMEDIA_LINK <b>Value 0x3E3</b>
FCodes.KBD_CREATE_MULTIMEDIA_LINK_TBL	KBD_CREATE_MULTIMEDIA_LINK_TBL <b>Value 0x3E4</b>
FCodes.KBD_MENUBARFOCUS	KBD_MENUBARFOCUS <b>Value 0x3F0</b>
FCodes.KBD_SINGLE_QUOTE	KBD_SINGLE_QUOTE <b>Value 0x400</b>
FCodes.KBD_DOUBLE_QUOTE	KBD_DOUBLE_QUOTE <b>Value 0x401</b>
FCodes.HIGH_CHAR_PREV	HIGH_CHAR_PREV <b>Value 0x410</b>
FCodes.HIGH_WORD_PREV	HIGH_WORD_PREV <b>Value 0x411</b>
FCodes.HIGH_LINE_PREV	HIGH_LINE_PREV <b>Value 0x412</b>
FCodes.HIGH_SENT_PREV	HIGH_SENT_PREV <b>Value 0x413</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.HIGH_PGF_PREV	HIGH_PGF_PREV <b>Value 0x414</b>
FCodes.HIGH_LINE_UP	HIGH_LINE_UP <b>Value 0x415</b>
FCodes.HIGH_LINE_DOWN	HIGH_LINE_DOWN <b>Value 0x416</b>
FCodes.HIGH_COL_TOP	HIGH_COL_TOP <b>Value 0x417</b>
FCodes.HIGH_COL_BOT	HIGH_COL_BOT <b>Value 0x418</b>
FCodes.HIGH_FLOW_BEG	HIGH_FLOW_BEG <b>Value 0x419</b>
FCodes.HIGH_FLOW_END	HIGH_FLOW_END <b>Value 0x41A</b>
FCodes.HIGH_LINE_BEG	HIGH_LINE_BEG <b>Value 0x41B</b>
FCodes.HIGH_LINE_END	HIGH_LINE_END <b>Value 0x41C</b>
FCodes.HIGH_ELEMENT_PREV	HIGH_ELEMENT_PREV <b>Value 0x420</b>
FCodes.HIGH_ELEMENT_NEXT	HIGH_ELEMENT_NEXT <b>Value 0x421</b>
FCodes.HIGH_SIBLINGS	HIGH_SIBLINGS <b>Value 0x422</b>
FCodes.HIGH_PARENT	HIGH_PARENT <b>Value 0x423</b>
FCodes.KBD_DFN_WEB_obsolete	KBD_DFN_WEB_obsolete <b>Value 0x500</b>
FCodes.KBD_DFN_LIB_obsolete	KBD_DFN_LIB_obsolete <b>Value 0x501</b>
FCodes.KBD_PASTE_RTF	KBD_PASTE_RTF <b>Value 0x502</b>
FCodes.KBD_PAGELAYOUT	KBD_PAGELAYOUT <b>Value 0x503</b>
FCodes.KBD_COPYRIGHT	KBD_COPYRIGHT <b>Value 0x504</b>



Property name	Data Type
FCodes.KBD_RESTOREFONT	KBD_RESTOREFONT <b>Value</b> 0x505
FCodes.KBD_SYMFONT	KBD_SYMFONT <b>Value</b> 0x506
FCodes.HIGH_CHAR_NEXT	HIGH_CHAR_NEXT <b>Value</b> 0x510
FCodes.HIGH_WORD_NEXT	HIGH_WORD_NEXT <b>Value</b> 0x511
FCodes.HIGH_LINE_NEXT	HIGH_LINE_NEXT <b>Value</b> 0x512
FCodes.HIGH_SENT_NEXT	HIGH_SENT_NEXT <b>Value</b> 0x513
FCodes.HIGH_PGF_NEXT	HIGH_PGF_NEXT <b>Value</b> 0x514
FCodes.HIGH_SAMECB	HIGH_SAMECB <b>Value</b> 0x515
FCodes.HIGH_SAMECOND	HIGH_SAMECOND <b>Value</b> 0x516
FCodes.HIGH_HYPertext	HIGH_HYPertext <b>Value</b> 0x517
FCodes.SEARCH_CB	SEARCH_CB <b>Value</b> 0x518
FCodes.SEARCH_CBTAG	SEARCH_CBTAG <b>Value</b> 0x519
FCodes.SEARCH_COND	SEARCH_COND <b>Value</b> 0x51A
FCodes.SEARCH_CBOVERRIDE	SEARCH_CBOVERRIDE <b>Value</b> 0x51B
FCodes.OBJ_SEL_FIRST	OBJ_SEL_FIRST <b>Value</b> 0x600
FCodes.OBJ_SEL_NEXT	OBJ_SEL_NEXT <b>Value</b> 0x601
FCodes.OBJ_SEL_EXTEND_NEXT	OBJ_SEL_EXTEND_NEXT <b>Value</b> 0x602
FCodes.OBJ_SEL_PREV	OBJ_SEL_PREV <b>Value</b> 0x603

<b>Property name</b>	<b>Data Type</b>
FCodes.OBJ_SEL_NEXT_WRAP	OBJ_SEL_NEXT_WRAP <b>Value 0x604</b>
FCodes.OBJ_SEL_PREV_WRAP	OBJ_SEL_PREV_WRAP <b>Value 0x605</b>
FCodes.OBJ_SEL_LAST	OBJ_SEL_LAST <b>Value 0x606</b>
FCodes.FOCUS_INPUT_DOC	FOCUS_INPUT_DOC <b>Value 0x620</b>
FCodes.FOCUS_INPUT_SEARCH	FOCUS_INPUT_SEARCH <b>Value 0x621</b>
FCodes.FOCUS_INPUT_MARKER	FOCUS_INPUT_MARKER <b>Value 0x622</b>
FCodes.FOCUS_INPUT_SPELL	FOCUS_INPUT_SPELL <b>Value 0x623</b>
FCodes.FOCUS_INPUT_HYPertext	FOCUS_INPUT_HYPertext <b>Value 0x624</b>
FCodes.FOCUS_INPUT_PGFFMT	FOCUS_INPUT_PGFFMT <b>Value 0x626</b>
FCodes.FOCUS_INPUT_FONTFMT	FOCUS_INPUT_FONTFMT <b>Value 0x627</b>
FCodes.FOCUS_INPUT_COND	FOCUS_INPUT_COND <b>Value 0x629</b>
FCodes.FOCUS_INPUT_CELLFMT	FOCUS_INPUT_CELLFMT <b>Value 0x62A</b>
FCodes.FOCUS_INPUT_CUSTRS	FOCUS_INPUT_CUSTRS <b>Value 0x62B</b>
FCodes.FOCUS_INPUT_TBLFMT	FOCUS_INPUT_TBLFMT <b>Value 0x62C</b>
FCodes.FOCUS_INPUT_STRWIN	FOCUS_INPUT_STRWIN <b>Value 0x62D</b>
FCodes.FOCUS_INPUT_VALIDATION	FOCUS_INPUT_VALIDATION <b>Value 0x62E</b>
FCodes.FOCUS_INPUT_ATTREDITOR	FOCUS_INPUT_ATTREDITOR <b>Value 0x62F</b>
FCodes.FOCUS_INPUT_ELEM_CTX	FOCUS_INPUT_ELEM_CTX <b>Value 0xA32</b>

## Object Reference

Property name	Data Type
FCodes.KBD_CLOSE_SEARCH	KBD_CLOSE_SEARCH <b>Value</b> 0x681
FCodes.KBD_CLOSE_MARKER	KBD_CLOSE_MARKER <b>Value</b> 0x682
FCodes.KBD_CLOSE_SPELL	KBD_CLOSE_SPELL <b>Value</b> 0x683
FCodes.KBD_CLOSE_PGFFMT	KBD_CLOSE_PGFFMT <b>Value</b> 0x684
FCodes.KBD_CLOSE_FONTFMT	KBD_CLOSE_FONTFMT <b>Value</b> 0x685
FCodes.KBD_CLOSE_COND	KBD_CLOSE_COND <b>Value</b> 0x686
FCodes.KBD_CLOSE_CUSTRS	KBD_CLOSE_CUSTRS <b>Value</b> 0x687
FCodes.KBD_CLOSE_TBLFMT	KBD_CLOSE_TBLFMT <b>Value</b> 0x688
FCodes.KBD_CLOSE_STRWIN	KBD_CLOSE_STRWIN <b>Value</b> 0x689
FCodes.KBD_CLOSE_VALIDATION	KBD_CLOSE_VALIDATION <b>Value</b> 0x68A
FCodes.KBD_CLOSE_ATTREDITOR	KBD_CLOSE_ATTREDITOR <b>Value</b> 0x68B
FCodes.KBD_CLOSE_ELEM_CTX	KBD_CLOSE_ELEM_CTX <b>Value</b> 0x68C
FCodes.KBD_CLOSE_HYPertext	KBD_CLOSE_HYPertext <b>Value</b> 0x68D
FCodes.KBD_PAUSE_1x00	KBD_PAUSE_1x00 <b>Value</b> 0x630
FCodes.KBD_PAUSE_0x10	KBD_PAUSE_0x10 <b>Value</b> 0x631
FCodes.KBD_PAUSE_0x01	KBD_PAUSE_0x01 <b>Value</b> 0x632
FCodes.KBD_SLOW_0x00	KBD_SLOW_0x00 <b>Value</b> 0x633
FCodes.KBD_SLOW_1x00	KBD_SLOW_1x00 <b>Value</b> 0x634

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_SLOW_0x10	KBD_SLOW_0x10 <b>Value</b> 0x635
FCodes.KBD_SLOW_0x01	KBD_SLOW_0x01 <b>Value</b> 0x636
FCodes.KBD_SET_TEXTFRAME_GRID	KBD_SET_TEXTFRAME_GRID <b>Value</b> 0x666
FCodes.KBD_PURIFY_NEW_LEAKS	KBD_PURIFY_NEW_LEAKS <b>Value</b> 0x650
FCodes.KBD_PURIFY_ALL_LEAKS	KBD_PURIFY_ALL_LEAKS <b>Value</b> 0x651
FCodes.KBD_PURIFY_CLEAR_LEAKS	KBD_PURIFY_CLEAR_LEAKS <b>Value</b> 0x652
FCodes.KBD_COUNT_SBLOCKS	KBD_COUNT_SBLOCKS <b>Value</b> 0x653
FCodes.KBD_API	KBD_API <b>Value</b> 0xDF0
FCodes.KBD_API_SHORTCUT	KBD_API_SHORTCUT <b>Value</b> 0xDF1
FCodes.TYPEIN	TYPEIN <b>Value</b> 0xDF2
FCodes.INLINE_TYPEIN	INLINE_TYPEIN <b>Value</b> 0xDF3
FCodes.KBD_VIEW_API	KBD_VIEW_API <b>Value</b> 0xDF4
FCodes.KBD_VIEW_API_SHORTCUT	KBD_VIEW_API_SHORTCUT <b>Value</b> 0xDF5
FCodes.KBD_VIEW_SWITCH	KBD_VIEW_SWITCH <b>Value</b> 0xDF6
FCodes.KBD_BACKSTACK	KBD_BACKSTACK <b>Value</b> 0xEF0
FCodes.KBD_UPSTACK	KBD_UPSTACK <b>Value</b> 0xEF1
FCodes.KBD_VIEWER	KBD_VIEWER <b>Value</b> 0xF00
FCodes.TOGGLE_FLUID_VIEW	TOGGLE_FLUID_VIEW <b>Value</b> 0x500

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_DSEXIT	KBD_DSEXIT <b>Value 0xF01</b>
FCodes.KBD_MEMFAIL	KBD_MEMFAIL <b>Value 0xF02</b>
FCodes.KBD_SAVEMETA	KBD_SAVEMETA <b>Value 0xF03</b>
FCodes.KBD_MEM_STATS	KBD_MEM_STATS <b>Value 0xF04</b>
FCodes.KBD_CACHE_STATS	KBD_CACHE_STATS <b>Value 0xF05</b>
FCodes.KBD_NEWVAR	KBD_NEWVAR <b>Value 0xF06</b>
FCodes.KBD_UPDATEREF	KBD_UPDATEREF <b>Value 0xF07</b>
FCodes.KBD_DEREFREF	KBD_DEREFREF <b>Value 0xF08</b>
FCodes.KBD_HEATREF	KBD_HEATREF <b>Value 0xF09</b>
FCodes.KBD_DOC_REPORT	KBD_DOC_REPORT <b>Value 0xF10</b>
FCodes.KBD_FULLRULERS	KBD_FULLRULERS <b>Value 0x3BD</b>
FCodes.KBD_ALLSELECT	KBD_ALLSELECT <b>Value 0xF20</b>
FCodes.KBD_OBJSELECT	KBD_OBJSELECT <b>Value 0xF21</b>
FCodes.KBD_TOOLWIN	KBD_TOOLWIN <b>Value 0xF22</b>
FCodes.KBD_PGFWIN	KBD_PGFWIN <b>Value 0xF23</b>
FCodes.KBD_FONTWIN	KBD_FONTWIN <b>Value 0xF24</b>
FCodes.KBD_RESIZEBOX	KBD_RESIZEBOX <b>Value 0xF25</b>
FCodes.KBD_MATHWIN	KBD_MATHWIN <b>Value 0xF26</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_RESIZEBOXM	KBD_RESIZEBOXM <b>Value 0xF27</b>
FCodes.KBD_ELEMENTWIN	KBD_ELEMENTWIN <b>Value 0xF28</b>
FCodes.KBD_SMALLTOOLWIN	KBD_SMALLTOOLWIN <b>Value 0xF29</b>
FCodes.KBD_OBJSELECT_NOPREF	KBD_OBJSELECT_NOPREF <b>Value 0xF2A</b>
FCodes.KBD_HOTSPOT	KBD_HOTSPOT <b>Value 0x4FD</b>
FCodes.KBD_EDITHOTSPOT	KBD_EDITHOTSPOT <b>Value 0x4FE</b>
FCodes.KBD_DELINKHOTSPOT	KBD_DELINKHOTSPOT <b>Value 0x507</b>
FCodes.KBD_ZOOMIN	KBD_ZOOMIN <b>Value 0xF30</b>
FCodes.KBD_ZOOMOUT	KBD_ZOOMOUT <b>Value 0xF31</b>
FCodes.KBD_ZOOM_FIT_PAGE	KBD_ZOOM_FIT_PAGE <b>Value 0xF32</b>
FCodes.KBD_ZOOM_FIT_WINDOW	KBD_ZOOM_FIT_WINDOW <b>Value 0xF33</b>
FCodes.KBD_ZOOM	KBD_ZOOM <b>Value 0xF34</b>
FCodes.KBD_ZOOM100	KBD_ZOOM100 <b>Value 0xF35</b>
FCodes.KBD_ZOOM_SET	KBD_ZOOM_SET <b>Value 0xF3C</b>
FCodes.KBD_ZOOM_FIT_TEXTFRAME	KBD_ZOOM_FIT_TEXTFRAME <b>Value 0xF3F</b>
FCodes.KBD_ZOOM_AUTOFIT_AFTER_ZOOM	KBD_ZOOM_AUTOFIT_AFTER_ZOOM <b>Value 0xF1A</b>
FCodes.KBD_TAGSTATUS	KBD_TAGSTATUS <b>Value 0xF36</b>
FCodes.KBD_PAGESTATUS	KBD_PAGESTATUS <b>Value 0xF37</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_HSCROLL	KBD_HSCROLL <b>Value 0xF38</b>
FCodes.KBD_VSCROLL	KBD_VSCROLL <b>Value 0xF39</b>
FCodes.KBD_RENAMEPAGE	KBD_RENAMEPAGE <b>Value 0xF3A</b>
FCodes.KBD_RENAMEFRAME	KBD_RENAMEFRAME <b>Value 0xF3B</b>
FCodes.KBD_FONTQUICK	KBD_FONTQUICK <b>Value 0xF40</b>
FCodes.KBD_PGFQUICK	KBD_PGFQUICK <b>Value 0xF41</b>
FCodes.KBD_VARQUICK	KBD_VARQUICK <b>Value 0xF42</b>
FCodes.KBD_CELLFMTQUICK	KBD_CELLFMTQUICK <b>Value 0xF43</b>
FCodes.KBD_CONDINQUICK	KBD_CONDINQUICK <b>Value 0xF44</b>
FCodes.KBD_CONDNOTINQUICK	KBD_CONDNOTINQUICK <b>Value 0xF45</b>
FCodes.KBD_UNCOND	KBD_UNCOND <b>Value 0xF46</b>
FCodes.KBD_CONDVISONLYQUICK	KBD_CONDVISONLYQUICK <b>Value 0xF47</b>
FCodes.KBD_INSERTQUICK	KBD_INSERTQUICK <b>Value 0xF48</b>
FCodes.KBD_WRAPQUICK	KBD_WRAPQUICK <b>Value 0xF49</b>
FCodes.KBD_CHANGEQUICK	KBD_CHANGEQUICK <b>Value 0xF4C</b>
FCodes.KBD_ATTREDITQUICK	KBD_ATTREDITQUICK <b>Value 0xF4D</b>
FCodes.PGF_DESIGN_CAT	PGF_DESIGN_CAT <b>Value 0xF4E</b>
FCodes.CHAR_DESIGN_CAT	CHAR_DESIGN_CAT <b>Value 0xF4F</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.TBL_DESIGN_CAT	TBL_DESIGN_CAT <b>Value 0xF5F</b>
FCodes.KBD_NOCHANGEDB	KBD_NOCHANGEDB <b>Value 0xF4A</b>
FCodes.KBD_RESETDB	KBD_RESETDB <b>Value 0xF4B</b>
FCodes.KBD_SMEQN	KBD_SMEQN <b>Value 0xF50</b>
FCodes.KBD_MEDEQN	KBD_MEDEQN <b>Value 0xF51</b>
FCodes.KBD_LGEQN	KBD_LGEQN <b>Value 0xF52</b>
FCodes.KBD_PUTINLINE	KBD_PUTINLINE <b>Value 0xF53</b>
FCodes.KBD_ANTIPUTINLINE	KBD_ANTIPUTINLINE <b>Value 0xF54</b>
FCodes.KBD_EVACUATE	KBD_EVACUATE <b>Value 0xF60</b>
FCodes.KBD_VERIFYCONTEXT	KBD_VERIFYCONTEXT <b>Value 0xF61</b>
FCodes.KBD_SAVEASDBRE	KBD_SAVEASDBRE <b>Value 0xF70</b>
FCodes.KBD_TEST_MODAL	KBD_TEST_MODAL <b>Value 0xF71</b>
FCodes.KBD_TEST_MODELESS	KBD_TEST_MODELESS <b>Value 0xF72</b>
FCodes.KBD_STUFF_ITEM	KBD_STUFF_ITEM <b>Value 0xF73</b>
FCodes.KBD_SAVESAS	KBD_SAVESAS <b>Value 0xF74</b>
FCodes.KBD_TEST_PRINTDBRE	KBD_TEST_PRINTDBRE <b>Value 0xF75</b>
FCodes.KBD_SAVEFMX	KBD_SAVEFMX <b>Value 0xF76</b>
FCodes.KBD_SAVEDBRE	KBD_SAVEDBRE <b>Value 0xF77</b>



<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_CAPTURE_LIVE_DIALOG	KBD_CAPTURE_LIVE_DIALOG <b>Value</b> 0xF78
FCodes.KBD_RM_MODE	KBD_RM_MODE <b>Value</b> 0xF79
FCodes.KBD_DRE_MODE_X	KBD_DRE_MODE_X <b>Value</b> 0xF7A
FCodes.KBD_DRE_MODE_W	KBD_DRE_MODE_W <b>Value</b> 0xF7B
FCodes.KBD_DRE_MODE_M	KBD_DRE_MODE_M <b>Value</b> 0xF7C
FCodes.KBD_TABLE_INS	KBD_TABLE_INS <b>Value</b> 0xF80
FCodes.KBD_TABLE_FORMAT	KBD_TABLE_FORMAT <b>Value</b> 0xF81
FCodes.KBD_TABLE_CELLFMT	KBD_TABLE_CELLFMT <b>Value</b> 0xF84
FCodes.KBD_TABLE_ROWFMFMT	KBD_TABLE_ROWFMFMT <b>Value</b> 0xF85
FCodes.KBD_TABLE_CUSTRS	KBD_TABLE_CUSTRS <b>Value</b> 0xF86
FCodes.KBD_TABLE_ADDRC	KBD_TABLE_ADDRC <b>Value</b> 0xF87
FCodes.KBD_TABLE_RESIZECOL	KBD_TABLE_RESIZECOL <b>Value</b> 0xF88
FCodes.KBD_TABLE_STRADDLE	KBD_TABLE_STRADDLE <b>Value</b> 0xF89
FCodes.KBD_TABLE_CONVERT	KBD_TABLE_CONVERT <b>Value</b> 0xF8A
FCodes.KBD_TABLE_DEBUG	KBD_TABLE_DEBUG <b>Value</b> 0xF8B
FCodes.KBD_TABLE_RULES	KBD_TABLE_RULES <b>Value</b> 0xF8C
FCodes.KBD_TABLE_EXIT_IP	KBD_TABLE_EXIT_IP <b>Value</b> 0xFC0
FCodes.KBD_TBL_DLG_UNIFY_TF	KBD_TBL_DLG_UNIFY_TF <b>Value</b> 0xF90

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_TBL_DLG_UNIFY_CF	KBD_TBL_DLG_UNIFY_CF <b>Value</b> 0xF91
FCodes.KBD_TBL_DLG_ADD_ABOVE	KBD_TBL_DLG_ADD_ABOVE <b>Value</b> 0xF92
FCodes.KBD_TBL_DLG_ADD_BELOW	KBD_TBL_DLG_ADD_BELOW <b>Value</b> 0xF93
FCodes.KBD_TBL_DLG_ADD_LEFT	KBD_TBL_DLG_ADD_LEFT <b>Value</b> 0xF94
FCodes.KBD_TBL_DLG_ADD_RIGHT	KBD_TBL_DLG_ADD_RIGHT <b>Value</b> 0xF95
FCodes.KBD_TBL_DLG_CLEAR_EMPTY	KBD_TBL_DLG_CLEAR_EMPTY <b>Value</b> 0xF96
FCodes.KBD_TBL_DLG_CLEAR_X	KBD_TBL_DLG_CLEAR_X <b>Value</b> 0xF97
FCodes.KBD_TBL_DLG_PASTE_REPL	KBD_TBL_DLG_PASTE_REPL <b>Value</b> 0xF98
FCodes.KBD_TBL_DLG_PASTE_BEFORE	KBD_TBL_DLG_PASTE_BEFORE <b>Value</b> 0xF99
FCodes.KBD_TBL_DLG_PASTE_AFTER	KBD_TBL_DLG_PASTE_AFTER <b>Value</b> 0xF9A
FCodes.KBD_TABLEWIN	KBD_TABLEWIN <b>Value</b> 0xFBB
FCodes.TBL_APPLY_CAT_TO_SEL	TBL_APPLY_CAT_TO_SEL <b>Value</b> 0xFBC
FCodes.KBD_CLOSE_TABLE_CATALOG	KBD_CLOSE_TABLE_CATALOG <b>Value</b> 0xFBD
FCodes.TBL_APPLY_TAG	TBL_APPLY_TAG <b>Value</b> 0xFBE
FCodes.KBD_COPYCELLFMT	KBD_COPYCELLFMT <b>Value</b> 0xF9B
FCodes.KBD_COPYCOLW	KBD_COPYCOLW <b>Value</b> 0xF9C
FCodes.KBD_COPYCOND	KBD_COPYCOND <b>Value</b> 0xF9D
FCodes.KBD_USEELTDEFSFROM	KBD_USEELTDEFSFROM <b>Value</b> 0xF9E

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_COPYATTRS	KBD_COPYATTRS <b>Value 0xF9F</b>
FCodes.KBD_ATOMIZE_INSET	KBD_ATOMIZE_INSET <b>Value 0xFAA</b>
FCodes.KBD_SWAP_RED_BLUE	KBD_SWAP_RED_BLUE <b>Value 0xFAB</b>
FCodes.KBD_REWRAP_INLINE_MATH	KBD_REWRAP_INLINE_MATH <b>Value 0xFAC</b>
FCodes.KBD_MODE_ROTATE_TOOL	KBD_MODE_ROTATE_TOOL <b>Value 0xFAD</b>
FCodes.KBD_XYZZY	KBD_XYZZY <b>Value 0xFED</b>
FCodes.KBD_XYZZZ	KBD_XYZZZ <b>Value 0xEE0</b>
FCodes.KBD_XYZZQ	KBD_XYZZQ <b>Value 0xEE1</b>
FCodes.KBD_ACTIVATE_IN_PDF_BYDEFAULT	KBD_ACTIVATE_IN_PDF_BYDEFAULT <b>Value 0x4FA</b>
FCodes.KBD_OPEN_IN_POPUP_WIN_IN_PDF	KBD_OPEN_IN_POPUP_WIN_IN_PDF <b>Value 0x4FB</b>
FCodes.KBD_GRAPHIC_SET_NAME	KBD_GRAPHIC_SET_NAME <b>Value 0x4FC</b>
FCodes.CMD_OPEN_DITAMAP_COMPONENT	CMD_OPEN_DITAMAP_COMPONENT <b>Value 0xF14</b>
FCodes.PGF_DESIGNKIT_APPLY	PGF_DESIGNKIT_APPLY <b>Value 0xAAA</b>
FCodes.CHAR_DESIGNKIT_APPLY	CHAR_DESIGNKIT_APPLY <b>Value 0xAAB</b>
FCodes.TBL_DESIGNKIT_APPLY	TBL_DESIGNKIT_APPLY <b>Value 0xAAC</b>
FCodes.PGF_DESIGNKIT_UPDATEALL	PGF_DESIGNKIT_UPDATEALL <b>Value 0xAAD</b>
FCodes.CHAR_DESIGNKIT_UPDATEALL	CHAR_DESIGNKIT_UPDATEALL <b>Value 0xAAE</b>
FCodes.TBL_DESIGNKIT_UPDATEALL	TBL_DESIGNKIT_UPDATEALL <b>Value 0xAAF</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.PGF_DESIGNKIT_NEWFORMAT	PGF_DESIGNKIT_NEWFORMAT <b>Value 0xB00</b>
FCodes.CHAR_DESIGNKIT_NEWFORMAT	CHAR_DESIGNKIT_NEWFORMAT <b>Value 0xB01</b>
FCodes.TBL_DESIGNKIT_NEWFORMAT	TBL_DESIGNKIT_NEWFORMAT <b>Value 0xB02</b>
FCodes.PGF_DESIGNKIT_UPDATEOPTIONS	PGF_DESIGNKIT_UPDATEOPTIONS <b>Value 0xB03</b>
FCodes.CHAR_DESIGNKIT_UPDATEOPTIONS	CHAR_DESIGNKIT_UPDATEOPTIONS <b>Value 0xB04</b>
FCodes.TBL_DESIGNKIT_UPDATEOPTIONS	TBL_DESIGNKIT_UPDATEOPTIONS <b>Value 0xB05</b>
FCodes.KBD_PGFFMT_DELETE	KBD_PGFFMT_DELETE <b>Value 0xB06</b>
FCodes.KBD_CHARFMT_DELETE	KBD_CHARFMT_DELETE <b>Value 0xB07</b>
FCodes.KBD_TBLFMT_DELETE	KBD_TBLFMT_DELETE <b>Value 0xB08</b>
FCodes.KBD_STYLEFMT_DELETE_UNUSED	KBD_STYLEFMT_DELETE_UNUSED <b>Value 0xB10</b>
FCodes.KBD_PGFFMT_DELETE_UNUSED	KBD_PGFFMT_DELETE_UNUSED <b>Value 0xB13</b>
FCodes.KBD_CHARFMT_DELETE_UNUSED	KBD_CHARFMT_DELETE_UNUSED <b>Value 0xB14</b>
FCodes.KBD_TBLFMT_DELETE_UNUSED	KBD_TBLFMT_DELETE_UNUSED <b>Value 0xB15</b>
FCodes.PGF_DESIGNKIT_RESET	PGF_DESIGNKIT_RESET <b>Value 0xAB0</b>
FCodes.CHAR_DESIGNKIT_RESET	CHAR_DESIGNKIT_RESET <b>Value 0xAB1</b>
FCodes.TBL_DESIGNKIT_RESET	TBL_DESIGNKIT_RESET <b>Value 0xAB2</b>
FCodes.KBD_STYLEFMT_DELETE	KBD_STYLEFMT_DELETE <b>Value 0xBB0</b>
FCodes.KBD_STYLE_CATALOG	KBD_STYLE_CATALOG <b>Value 0xBB1</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_STYLE_DESIGNER	KBD_STYLE_DESIGNER <b>Value 0xBB2</b>
FCodes.CONDTEXT_KIT_APPLY	CONDTEXT_KIT_APPLY <b>Value 0xB09</b>
FCodes.ALT_TXT_INSET	ALT_TXT_INSET <b>Value 0xB11</b>
FCodes.ALT_GFX_INSET	ALT_GFX_INSET <b>Value 0xB12</b>
FCodes.KBD_TBLSEL_CELL	KBD_TBLSEL_CELL <b>Value 0xFA0</b>
FCodes.KBD_TBLSEL_ROW	KBD_TBLSEL_ROW <b>Value 0xFA1</b>
FCodes.KBD_TBLSEL_COL	KBD_TBLSEL_COL <b>Value 0xFA2</b>
FCodes.KBD_TBLSEL_TABLE	KBD_TBLSEL_TABLE <b>Value 0xFA3</b>
FCodes.KBD_TBLSEL_CELLTEXT	KBD_TBLSEL_CELLTEXT <b>Value 0xFA4</b>
FCodes.KBD_TBLSEL_COLBODY	KBD_TBLSEL_COLBODY <b>Value 0xFA5</b>
FCodes.KBD_TBLIP_TOPLEFT	KBD_TBLIP_TOPLEFT <b>Value 0xFB0</b>
FCodes.KBD_TBLIP_RIGHT	KBD_TBLIP_RIGHT <b>Value 0xFB1</b>
FCodes.KBD_TBLIP_LEFT	KBD_TBLIP_LEFT <b>Value 0xFB2</b>
FCodes.KBD_TBLIP_ABOVE	KBD_TBLIP_ABOVE <b>Value 0xFB3</b>
FCodes.KBD_TBLIP_BELOW	KBD_TBLIP_BELOW <b>Value 0xFB4</b>
FCodes.KBD_TBLIP_LEFTMOST	KBD_TBLIP_LEFTMOST <b>Value 0xFB5</b>
FCodes.KBD_TBLIP_RIGHTMOST	KBD_TBLIP_RIGHTMOST <b>Value 0xFB6</b>
FCodes.KBD_TBLIP_NEXT	KBD_TBLIP_NEXT <b>Value 0xFB7</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_TBLIP_PREV	KBD_TBLIP_PREV <b>Value 0xFB8</b>
FCodes.KBD_TBLIP_TOP	KBD_TBLIP_TOP <b>Value 0xFB9</b>
FCodes.KBD_TBLIP_BOTTOM	KBD_TBLIP_BOTTOM <b>Value 0xFBA</b>
FCodes.PGF_KBD_SIDEBODY	PGF_KBD_SIDEBODY <b>Value 0xFC9</b>
FCodes.PGF_KBD_SIDEHEAD_FIRST_BASELINE	PGF_KBD_SIDEHEAD_FIRST_BASELINE <b>Value 0xFCA</b>
FCodes.PGF_KBD_SIDEHEAD_LAST_BASELINE	PGF_KBD_SIDEHEAD_LAST_BASELINE <b>Value 0xFCB</b>
FCodes.PGF_KBD_SIDEHEAD_TOP	PGF_KBD_SIDEHEAD_TOP <b>Value 0xFCC</b>
FCodes.PGF_KBD_RUN_IN	PGF_KBD_RUN_IN <b>Value 0xFCD</b>
FCodes.PGF_KBD_FULLSTRADDLE	PGF_KBD_FULLSTRADDLE <b>Value 0xFCE</b>
FCodes.PGF_KBD_BODYSTRADDLE	PGF_KBD_BODYSTRADDLE <b>Value 0xFCF</b>
FCodes.KBD_TBL_DLG_SHRINKWRAP	KBD_TBL_DLG_SHRINKWRAP <b>Value 0xFD0</b>
FCodes.ELEM_INS_CAT_AT_SEL	ELEM_INS_CAT_AT_SEL <b>Value 0xA10</b>
FCodes.ELEM_WRAP_CAT_AT_SEL	ELEM_WRAP_CAT_AT_SEL <b>Value 0xA11</b>
FCodes.ELEM_CHANGE_CAT_AT_SEL	ELEM_CHANGE_CAT_AT_SEL <b>Value 0xA12</b>
FCodes.ELEM_INSERT_CUSTOM_1	ELEM_INSERT_CUSTOM_1 <b>Value 0xA21</b>
FCodes.ELEM_INSERT_CUSTOM_2	ELEM_INSERT_CUSTOM_2 <b>Value 0xA22</b>
FCodes.ELEM_INSERT_CUSTOM_3	ELEM_INSERT_CUSTOM_3 <b>Value 0xA23</b>
FCodes.ELEM_INSERT_CUSTOM_4	ELEM_INSERT_CUSTOM_4 <b>Value 0xA24</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.ELEM_INSERT_CUSTOM_5	ELEM_INSERT_CUSTOM_5 <b>Value</b> 0xA25
FCodes.ELEM_INSERT_CUSTOM_6	ELEM_INSERT_CUSTOM_6 <b>Value</b> 0xA26
FCodes.ELEM_INSERT_CUSTOM_7	ELEM_INSERT_CUSTOM_7 <b>Value</b> 0xA27
FCodes.ELEM_INSERT_CUSTOM_8	ELEM_INSERT_CUSTOM_8 <b>Value</b> 0xA28
FCodes.ELEM_INSERT_CUSTOM_9	ELEM_INSERT_CUSTOM_9 <b>Value</b> 0xA29
FCodes.ELEM_INSERT_CUSTOM_10	ELEM_INSERT_CUSTOM_10 <b>Value</b> 0xA2A
FCodes.KBD_STRIP_FLOWSTRUCTURE	KBD_STRIP_FLOWSTRUCTURE <b>Value</b> 0xA30
FCodes.KBD_SHOW_ELEM_CTX	KBD_SHOW_ELEM_CTX <b>Value</b> 0xA31
FCodes.KBD_NORMALIZE_TAGS	KBD_NORMALIZE_TAGS <b>Value</b> 0xA33
FCodes.KBD_STRWIN	KBD_STRWIN <b>Value</b> 0xFD1
FCodes.KBD_ELEM_BORDER	KBD_ELEM_BORDER <b>Value</b> 0xFD2
FCodes.KBD_ELEM_MERGE_1ST	KBD_ELEM_MERGE_1ST <b>Value</b> 0xFD3
FCodes.KBD_ELEM_MERGE_LAST	KBD_ELEM_MERGE_LAST <b>Value</b> 0xFD4
FCodes.KBD_ELEM_SPLIT	KBD_ELEM_SPLIT <b>Value</b> 0xFD5
FCodes.KBD_ELEM_UNWRAP	KBD_ELEM_UNWRAP <b>Value</b> 0xFD6
FCodes.KBD_ELEM_CAT_OPTS	KBD_ELEM_CAT_OPTS <b>Value</b> 0xFD7
FCodes.KBD_SETELCATSTRICT	KBD_SETELCATSTRICT <b>Value</b> 0xFD8
FCodes.KBD_SETELCATLOOSE	KBD_SETELCATLOOSE <b>Value</b> 0xFD9

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_SETELCATCHILD	KBD_SETELCATCHILD <b>Value</b> 0xFDA
FCodes.KBD_SETELCATALL	KBD_SETELCATALL <b>Value</b> 0xFDB
FCodes.KBD_SETELCATFREQ	KBD_SETELCATFREQ <b>Value</b> 0xFF7
FCodes.ELEM_BAM	ELEM_BAM <b>Value</b> 0xFDC
FCodes.KBD_ELEM_TAGS	KBD_ELEM_TAGS <b>Value</b> 0xFDD
FCodes.KBD_BANNER_TEXT	KBD_BANNER_TEXT <b>Value</b> 0x137
FCodes.KBD_ELEM_PROMOTE	KBD_ELEM_PROMOTE <b>Value</b> 0xFDE
FCodes.KBD_ELEM_DEMOTE	KBD_ELEM_DEMOTE <b>Value</b> 0xFDF
FCodes.KBD_VALIDATION	KBD_VALIDATION <b>Value</b> 0xFE0
FCodes.KBD_VAL_ELEM	KBD_VAL_ELEM <b>Value</b> 0xFE1
FCodes.KBD_VAL_FLOW	KBD_VAL_FLOW <b>Value</b> 0xFE2
FCodes.KBD_VAL_DOC	KBD_VAL_DOC <b>Value</b> 0xFE3
FCodes.KBD_VAL_IGNORE	KBD_VAL_IGNORE <b>Value</b> 0xFE4
FCodes.KBD_VAL_START	KBD_VAL_START <b>Value</b> 0xFE5
FCodes.KBD_VAL_ALLOW	KBD_VAL_ALLOW <b>Value</b> 0xFE6
FCodes.KBD_VAL_CLEAR	KBD_VAL_CLEAR <b>Value</b> 0xFE7
FCodes.SW_CSR_UP	SW_CSR_UP <b>Value</b> 0xFE8
FCodes.SW_CSR_DOWN	SW_CSR_DOWN <b>Value</b> 0xFE9



<b>Property name</b>	<b>Data Type</b>
FCodes.SW_CSR_RIGHT	SW_CSR_RIGHT <b>Value 0xFEA</b>
FCodes.SW_CSR_LEFT	SW_CSR_LEFT <b>Value 0xFEB</b>
FCodes.KBD_INLINE_ATTREDTR	KBD_INLINE_ATTREDTR <b>Value 0x29B</b>
FCodes.KBD_ATTRIBUTE_EDIT	KBD_ATTRIBUTE_EDIT <b>Value 0xFEC</b>
FCodes.KBD_ATTR_CONFIG_FILE_MAKER	KBD_ATTR_CONFIG_FILE_MAKER <b>Value 0xFFA</b>
FCodes.KBD_ATTR_DISP_OPTS	KBD_ATTR_DISP_OPTS <b>Value 0xFEE</b>
FCodes.KBD_FB_EDIT_PREFS	KBD_FB_EDIT_PREFS <b>Value 0xFEFF</b>
FCodes.KBD_REMOVE_STRUCT	KBD_REMOVE_STRUCT <b>Value 0xFF0</b>
FCodes.KBD_TOGGLE_INCLUSION	KBD_TOGGLE_INCLUSION <b>Value 0xFF1</b>
FCodes.KBD_TOGGLE_COLLAPSE	KBD_TOGGLE_COLLAPSE <b>Value 0xFF2</b>
FCodes.KBD_TOGGLE_COLLAPSE_ALL	KBD_TOGGLE_COLLAPSE_ALL <b>Value 0xFF3</b>
FCodes.KBD_TOGGLE_COLLAPSE_CHILDREN	KBD_TOGGLE_COLLAPSE_CHILDREN <b>Value 0xA0D</b>
FCodes.KBD_ELEM_TRANSPOSE_PREV	KBD_ELEM_TRANSPOSE_PREV <b>Value 0xFF4</b>
FCodes.KBD_ELEM_TRANSPOSE_NEXT	KBD_ELEM_TRANSPOSE_NEXT <b>Value 0xFF5</b>
FCodes.KBD_VAL_IGNORE_ATTR	KBD_VAL_IGNORE_ATTR <b>Value 0xFF6</b>
FCodes.KBD_NAMESPACES	KBD_NAMESPACES <b>Value 0xFF8</b>
FCodes.KBD_TOGGLE_STRUCT_AND_DOCWIN	KBD_TOGGLE_STRUCT_AND_DOCWIN <b>Value 0xFF9</b>
FCodes.FM_SEPARATOR	FM_SEPARATOR <b>Value 0xF12</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.FM_TERMINATE	FM_TERMINATE <b>Value 0xFF</b>
FCodes.FM_RPT_CMDS_BY_TAG	FM_RPT_CMDS_BY_TAG <b>Value 0xD00</b>
FCodes.FM_RPT_CMDS_BY_SHORTCUT	FM_RPT_CMDS_BY_SHORTCUT <b>Value 0xD01</b>
FCodes.KBD_PASTESPECIAL	KBD_PASTESPECIAL <b>Value 0xD10</b>
FCodes.KBD_CLOSE_TOP_PALETTE	KBD_CLOSE_TOP_PALETTE <b>Value 0xD20</b>
FCodes.KBD_CLOSE_PARAGRAPH_CATALOG	KBD_CLOSE_PARAGRAPH_CATALOG <b>Value 0xD21</b>
FCodes.KBD_CLOSE_CHARACTER_CATALOG	KBD_CLOSE_CHARACTER_CATALOG <b>Value 0xD22</b>
FCodes.KBD_CLOSE_TOOLS_PALETTE	KBD_CLOSE_TOOLS_PALETTE <b>Value 0xD23</b>
FCodes.KBD_CLOSE_EQUATIONS_PALETTE	KBD_CLOSE_EQUATIONS_PALETTE <b>Value 0xD24</b>
FCodes.KBD_CLOSE_ELEMENT_CATALOG	KBD_CLOSE_ELEMENT_CATALOG <b>Value 0xD26</b>
FCodes.KBD_CLOSE_PARAGRAPH_DESIGNER	KBD_CLOSE_PARAGRAPH_DESIGNER <b>Value 0xD28</b>
FCodes.KBD_CLOSE_CHARACTER_DESIGNER	KBD_CLOSE_CHARACTER_DESIGNER <b>Value 0xD29</b>
FCodes.KBD_CLOSE_TABLE_DESIGNER	KBD_CLOSE_TABLE_DESIGNER <b>Value 0xD2A</b>
FCodes.KBD_WINDOWFULL_UP	KBD_WINDOWFULL_UP <b>Value 0xD40</b>
FCodes.KBD_WINDOWFULL_DOWN	KBD_WINDOWFULL_DOWN <b>Value 0xD41</b>
FCodes.KBD_VIEWER_MAKE_DOCUMENT	KBD_VIEWER_MAKE_DOCUMENT <b>Value 0xD50</b>
FCodes.KBD_VIEWER_MAKE_PALETTE	KBD_VIEWER_MAKE_PALETTE <b>Value 0xD51</b>
FCodes.KBD_VIEWER_MAKE_MODELESSDIALOG	KBD_VIEWER_MAKE_MODELESSDIALOG <b>Value 0xD52</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_Item1stLogical	KBD_Item1stLogical <b>Value 0xD60</b>
FCodes.KBD_ItemNextLogical	KBD_ItemNextLogical <b>Value 0xD61</b>
FCodes.KBD_ItemPrevLogical	KBD_ItemPrevLogical <b>Value 0xD62</b>
FCodes.KBD_ItemNextPhysical	KBD_ItemNextPhysical <b>Value 0xD63</b>
FCodes.KBD_ItemPrevPhysical	KBD_ItemPrevPhysical <b>Value 0xD64</b>
FCodes.KBD_ItemLeft	KBD_ItemLeft <b>Value 0xD65</b>
FCodes.KBD_ItemRight	KBD_ItemRight <b>Value 0xD66</b>
FCodes.KBD_ItemUp	KBD_ItemUp <b>Value 0xD67</b>
FCodes.KBD_ItemDown	KBD_ItemDown <b>Value 0xD68</b>
FCodes.KBD_ItemSelect	KBD_ItemSelect <b>Value 0xD69</b>
FCodes.KBD_ItemFirstFocus	KBD_ItemFirstFocus <b>Value 0xD6A</b>
FCodes.INIT_DIALOG	INIT_DIALOG <b>Value 0xD80</b>
FCodes.KBD_RENAMEORPLAIN	KBD_RENAMEORPLAIN <b>Value 0x900</b>
FCodes.KBD_WIN_CASCADE	KBD_WIN_CASCADE <b>Value 0x903</b>
FCodes.KBD_WIN_TILE	KBD_WIN_TILE <b>Value 0x904</b>
FCodes.KBD_TAB_LEFT	KBD_TAB_LEFT <b>Value 0x906</b>
FCodes.KBD_TAB_CENTER	KBD_TAB_CENTER <b>Value 0x907</b>
FCodes.KBD_TAB_RIGHT	KBD_TAB_RIGHT <b>Value 0x908</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_TAB_DECIMAL	KBD_TAB_DECIMAL <b>Value 0x909</b>
FCodes.KBD_PRINTSETUP	KBD_PRINTSETUP <b>Value 0x90A</b>
FCodes.KBD_MINIMIZE	KBD_MINIMIZE <b>Value 0x90F</b>
FCodes.KBD_MAXIMIZE	KBD_MAXIMIZE <b>Value 0x910</b>
FCodes.KBD_RESTORE	KBD_RESTORE <b>Value 0x911</b>
FCodes.KBD_EDITLINKS	KBD_EDITLINKS <b>Value 0x912</b>
FCodes.KBD_INSERTOBJECT	KBD_INSERTOBJECT <b>Value 0x913</b>
FCodes.KBD_CONTEXTMENU	KBD_CONTEXTMENU <b>Value 0x914</b>
FCodes.KBD_EMBEDDED00	KBD_EMBEDDED00 <b>Value 0x920</b>
FCodes.KBD_EMBEDDED01	KBD_EMBEDDED01 <b>Value 0x921</b>
FCodes.KBD_EMBEDDED02	KBD_EMBEDDED02 <b>Value 0x922</b>
FCodes.KBD_EMBEDDED03	KBD_EMBEDDED03 <b>Value 0x923</b>
FCodes.KBD_EMBEDDED04	KBD_EMBEDDED04 <b>Value 0x924</b>
FCodes.KBD_EMBEDDED05	KBD_EMBEDDED05 <b>Value 0x925</b>
FCodes.KBD_EMBEDDED06	KBD_EMBEDDED06 <b>Value 0x926</b>
FCodes.KBD_EMBEDDED07	KBD_EMBEDDED07 <b>Value 0x927</b>
FCodes.KBD_EMBEDDED08	KBD_EMBEDDED08 <b>Value 0x928</b>
FCodes.KBD_EMBEDDED09	KBD_EMBEDDED09 <b>Value 0x92A</b>

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_EMBEDDED10	KBD_EMBEDDED10 <b>Value</b> 0x92B
FCodes.KBD_EMBEDDED11	KBD_EMBEDDED11 <b>Value</b> 0x92C
FCodes.KBD_EMBEDDED12	KBD_EMBEDDED12 <b>Value</b> 0x92D
FCodes.KBD_EMBEDDED13	KBD_EMBEDDED13 <b>Value</b> 0x92E
FCodes.KBD_EMBEDDED14	KBD_EMBEDDED14 <b>Value</b> 0x92F
FCodes.KBD_EMBEDDED15	KBD_EMBEDDED15 <b>Value</b> 0x930
FCodes.KBD_EMBEDDED16	KBD_EMBEDDED16 <b>Value</b> 0x931
FCodes.KBD_EMBEDDED17	KBD_EMBEDDED17 <b>Value</b> 0x932
FCodes.KBD_EMBEDDED18	KBD_EMBEDDED18 <b>Value</b> 0x933
FCodes.KBD_EMBEDDED19	KBD_EMBEDDED19 <b>Value</b> 0x934
FCodes.KBD_EMBEDDEDMAX	KBD_EMBEDDEDMAX <b>Value</b> 0x94F
FCodes.KBD_GOTOLINEN	KBD_GOTOLINEN <b>Value</b> 0x970
FCodes.KBD_SHOWNEXT	KBD_SHOWNEXT <b>Value</b> 0x971
FCodes.KBD_SHOWPREV	KBD_SHOWPREV <b>Value</b> 0x972
FCodes.KBD_SCREENMODE_TOGGLE	KBD_SCREENMODE_TOGGLE <b>Value</b> 0x978
FCodes.KBD_SCREENMODE_STANDARD	KBD_SCREENMODE_STANDARD <b>Value</b> 0x979
FCodes.KBD_SCREENMODE_FULLSCREEN_UI	KBD_SCREENMODE_FULLSCREEN_UI <b>Value</b> 0x97A
FCodes.KBD_SCREENMODE_FULLSCREEN	KBD_SCREENMODE_FULLSCREEN <b>Value</b> 0x97B

<b>Property name</b>	<b>Data Type</b>
FCodes.KBD_UI_PREFERENCE	KBD_UI_PREFERENCE <b>Value</b> 0x980
FCodes.KBD_UIALERTSTRINGS_PREF	KBD_UIALERTSTRINGS_PREF <b>Value</b> 0x981
FCodes.KBD_TOOLBAR_SHOWALL	KBD_TOOLBAR_SHOWALL <b>Value</b> 0x989
FCodes.KBD_TOOLBAR_HIDEALL	KBD_TOOLBAR_HIDEALL <b>Value</b> 0x98A
FCodes.KBD_TOOLBAR_BASE	KBD_TOOLBAR_BASE <b>Value</b> 0x98F
FCodes.KBD_BROWSE_URL	KBD_BROWSE_URL <b>Value</b> 0x9A0
FCodes.KBD_MATH_BASE	KBD_MATH_BASE <b>Value</b> 0x1000
FCodes.TEXTSEL_QUICK_COPY	TEXTSEL_QUICK_COPY <b>Value</b> 0x0001
FCodes.TEXTSEL_EXTEND	TEXTSEL_EXTEND <b>Value</b> 0x0002
FCodes.TEXTSEL_EXTEND_WORD	TEXTSEL_EXTEND_WORD <b>Value</b> 0x0004
FCodes.TEXTSEL_EXTEND_LINE	TEXTSEL_EXTEND_LINE <b>Value</b> 0x0008
FCodes.TEXTSEL_EXTEND_PGF	TEXTSEL_EXTEND_PGF <b>Value</b> 0x0010
FCodes.TEXTSEL_WORD	TEXTSEL_WORD <b>Value</b> 0x0020
FCodes.TEXTSEL_LINE	TEXTSEL_LINE <b>Value</b> 0x0040
FCodes.TEXTSEL_PGF	TEXTSEL_PGF <b>Value</b> 0x0080
FCodes.TEXTSEL_SELECT_ONLY	TEXTSEL_SELECT_ONLY <b>Value</b> 0x0100
FCodes.TEXTSEL_EXTEND_SENT	TEXTSEL_EXTEND_SENT <b>Value</b> 0x0200
FCodes.TEXTSEL_SENT	TEXTSEL_SENT <b>Value</b> 0x0400

<b>Property name</b>	<b>Data Type</b>
FCodes.TEXTSEL_EXTEND_ELEMENT	TEXTSEL_EXTEND_ELEMENT <b>Value</b> 0x0800
FCodes.TEXTSEL_ELEMENT	TEXTSEL_ELEMENT <b>Value</b> 0x1000
FCodes.TEXTSEL_DRAGGING	TEXTSEL_DRAGGING <b>Value</b> 0x2000
FCodes.EXTEND_SEL	EXTEND_SEL <b>Value</b> (0x0002 0x0004 0x0008 0x0010 0x0800)
FCodes.MIF_SAVE_TEXT	MIF_SAVE_TEXT <b>Value</b> 0x000001
FCodes.MIF_SAVE_TAGS	MIF_SAVE_TAGS <b>Value</b> 0x000002
FCodes.MIF_SAVE_FMTS	MIF_SAVE_FMTS <b>Value</b> 0x000004
FCodes.MIF_SAVE_FONTS	MIF_SAVE_FONTS <b>Value</b> 0x000008
FCodes.MIF_SAVE_MKRS	MIF_SAVE_MKRS <b>Value</b> 0x000010
FCodes.MIF_SAVE_AFMS	MIF_SAVE_AFMS <b>Value</b> 0x000020
FCodes.MIF_SAVE_LAYT	MIF_SAVE_LAYT <b>Value</b> 0x000040
FCodes.MIF_SAVE_MPAGE	MIF_SAVE_MPAGE <b>Value</b> 0x000080
FCodes.MIF_SAVE_FCAT	MIF_SAVE_FCAT <b>Value</b> 0x000100
FCodes.MIF_SAVE_PCAT	MIF_SAVE_PCAT <b>Value</b> 0x000200
FCodes.MIF_SAVE_CCAT	MIF_SAVE_CCAT <b>Value</b> 0x000400
FCodes.MIF_SAVE_TMPLT	MIF_SAVE_TMPLT <b>Value</b> 0x000800
FCodes.MIF_SAVE_DICT	MIF_SAVE_DICT <b>Value</b> 0x001000
FCodes.MIF_SAVE_VARS	MIF_SAVE_VARS <b>Value</b> 0x002000

<b>Property name</b>	<b>Data Type</b>
FCodes.MIF_SAVE_TABLECATS	MIF_SAVE_TABLECATS <b>Value</b> 0x004000
FCodes.MIF_SAVE_TABLES	MIF_SAVE_TABLES <b>Value</b> 0x008000
FCodes.MIF_SAVE_ECAT	MIF_SAVE_ECAT <b>Value</b> 0x010000
FCodes.MIF_SAVE_ELEMENTS	MIF_SAVE_ELEMENTS <b>Value</b> 0x020000
FCodes.MIF_SAVE_COLORCAT	MIF_SAVE_COLORCAT <b>Value</b> 0x040000
FCodes.MIF_SAVE_VIEWSET	MIF_SAVE_VIEWSET <b>Value</b> 0x080000
FCodes.MIF_SAVE_DATA LINKS	MIF_SAVE_DATA LINKS <b>Value</b> 0x100000
FCodes.MIF_SAVE_FPLCAT	MIF_SAVE_FPLCAT <b>Value</b> 0x200000
FCodes.MIF_SAVE_TEXTINSETS	MIF_SAVE_TEXTINSETS <b>Value</b> 0x400000
FCodes.MIF_SAVE_RUBIS	MIF_SAVE_RUBIS <b>Value</b> 0x800000
FCodes.MIF_SAVE_BANNERTXT	MIF_SAVE_BANNERTXT <b>Value</b> 0x1000000
FCodes.KBD_OPEN_REPOSITORY	KBD_OPEN_REPOSITORY <b>Value</b> 0x801
FCodes.KBD_CMS_CONNECTION_MGR	KBD_CMS_CONNECTION_MGR <b>Value</b> 0x802
FCodes.KBD_NEW_XML	KBD_NEW_XML <b>Value</b> 0x803
FCodes.KBD_CMS_UPLOAD_DOC_OR_BOOK	KBD_CMS_UPLOAD_DOC_OR_BOOK <b>Value</b> 0x805
FCodes.KBD_REFRESH_DITAMAP_RMVIEW	KBD_REFRESH_DITAMAP_RMVIEW <b>Value</b> 0x807
FCodes.STYLE_APPLY_CAT_TO_SEL	STYLE_APPLY_CAT_TO_SEL <b>Value</b> 0x808



Property name	Data Type
FCodes.KBD_SHOW_ELEM_CTX_AV	KBD_SHOW_ELEM_CTX_AV <b>Value</b> 0x809
FCodes.FOCUS_INPUT_ELEM_CTX_AV	FOCUS_INPUT_ELEM_CTX_AV <b>Value</b> 0x810
FCodes.KBD_CLOSE_ELEM_CTX_AV	KBD_CLOSE_ELEM_CTX_AV <b>Value</b> 0x811

## FilterArgs

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

## FilterArgs methods

[FilterArgs.](#)

## Flow

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

Property name	Data Type	Description
MinHang	int	Maximum character height for synchronization of first line in column. If characters exceed this height, FrameMaker does not synchronize the first line.
Name	string	Name of flow tag.
NextFlowInDoc	<a href="#">Flow</a>	Next flow in document (FO_Flow ID).
SideHeadRoomInFlow	int	1 if <code>Leave Room for Sideheads in Flow</code> is enabled; 0 otherwise.
Spacing	int	Line spacing for synchronized baselines.
HighestLevelElement	<a href="#">Element</a>	Highest-level element in flow (FO_Element ID)

## Flow methods

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

## FmtChangeList

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

Property name	Data Type	Description
MinSpace	int	Minimum word spacing (percentage of an em space in current font).
OptSpace	int	Optimum word spacing.
TopSeparator	string	Name of frame to put above paragraph.
TopSepAtIndent	int	1 if the position of the frame specified by TopSeparator is at the current left indent; 0 otherwise.
MinJRomSpace	int	Minimum Asian-Roman space.
OptJRomSpace	int	Optimum Asian-Roman space.
MaxJRomSpace	int	Maximum Asian-Roman space.
MinJLetSpace	int	Minimum Asian letter space.
OptJLetSpace	int	Optimum Asian letter space.
MaxJLetSpace	int	Maximum Asian letter space.
YakumonoType	int	Denotes the The Yakumono rules to handle punctuation characters. The value can be one of: <ul style="list-style-type: none"> <li>• Constants.FV_FLOATING_YAKUMONO (0x00)</li> <li>• Constants.FV_MONOSPACE_YAKUMONO (0x01)</li> <li>• Constants.FV_FIXED_YAKUMONO (0x02)</li> </ul>
AutoNumChar	string	Character format for the automatic numbering string specified by AutoNumString; " " if the default character format is used.
AutoNumString	string	Autonumber format string (for example, <n>.<n+>)
NumAtEnd	int	1 if the numbering position is at the End of the Paragraph; 0 if it is at the Beginning of the Paragraph.
PgflsAutoNum	int	1 if autonumbering is enabled; 0 otherwise.
FirstIndent	int	The paragraph's first-line left margin, measured from the left side of the current text column (0 cm to 100 cm).
FirstIndentChange	int	Amount by which to increase or decrease the first-line left margin.
FirstIndentIsRelative	int	1 if the first indent is relative to the left indent.
FirstIndentRelPos	int	Position relative to left indent if FirstIndentIsRelative is True.
Leading	int	Denotes the space below each line in the paragraph.
LeadingChange	int	Amount by which to increase or decrease the leading.
LeftIndent	int	The paragraph's left margin, measured from the left side of the current text column (0 cm to 100 cm).
LeftIndentChange	int	Amount by which to increase or decrease the left margin.
LineSpacingFixed	int	1 if the line spacing is fixed.
MoveTabs	int	Amount by which to move all tab positions in the paragraph.

Property name	Data Type	Description
NumTabs	int	The number of tabs in the paragraph. To clear all the tabs in the paragraph, set NumTabs to 0.
PgfAlignment	int	Horizontal alignment of the paragraph. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PGF_LEFT (0x01)</li> <li>• Constants.FV_PGF_RIGHT (0x02)</li> <li>• Constants.FV_PGF_CENTER (0x03)</li> <li>• Constants.FV_PGF_JUSTIFIED (0x04)</li> </ul>
RightIndent	int	The paragraph's right margin, measured from the right side of the current text column.
RightIndentChange	int	Amount by which to increase or decrease the right margin.
SpaceAbove	int	Indicates the space above the paragraph.
SpaceAboveChange	int	Amount by which to increase or decrease the space above.
SpaceBelow	int	Indicates the space below the paragraph.
SpaceBelowChange	int	Amount by which to increase or decrease the space below.
Tabs	<a href="#">Tabs</a>	An array of tab descriptions that specify the positions and types of tab stops in the paragraph.
Capitalization	int	Type of capitalization to use. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_CAPITAL_CASE_NORM ()</li> <li>• Constants.FV_CAPITAL_CASE_SMALL ()</li> <li>• Constants.FV_CAPITAL_CASE_LOWER ()</li> <li>• Constants.FV_CAPITAL_CASE_UPPER ()</li> </ul>
ChangeBar	int	1 if Change Bars are on;0 otherwise.
Color	<a href="#">Color</a>	The spot color (FO_Color ID).
CombinedFontFamily	<a href="#">CombinedFontDefn</a>	The combined font definition (FO_CombinedFontDefn)
FontAngle	int	Denotes the font angle (specifies an index into the array of font angles provided by the session property FontAngleNames).
FontFamily	int	Denotes the font family (specifies an index into the array of font families provided by the session property FontFamilyNames).

Property name	Data Type	Description
Language	int	<p>The hyphenation and spell-checking language to use. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_LANG_BRAZILIAN (0x0B)</li> <li>• Constants.FV_LANG_BRITISH (0x02)</li> <li>• Constants.FV_LANG_CANADIAN_FRENCH (0x06)</li> <li>• Constants.FV_LANG_CATALAN (0x08)</li> <li>• Constants.FV_LANG_DANISH (0x0C)</li> <li>• Constants.FV_LANG_DUTCH (0x0D)</li> <li>• Constants.FV_LANG_ENGLISH (0x01)</li> <li>• Constants.FV_LANG_FINNISH (0x10)</li> <li>• Constants.FV_LANG_FRENCH (0x05)</li> <li>• Constants.FV_LANG_GERMAN (0x03)</li> <li>• Constants.FV_LANG_ITALIAN (0x09)</li> <li>• Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li>• Constants.FV_LANG_NORWEGIAN (0x0E)</li> <li>• Constants.FV_LANG_NYNORSK (0x0F)</li> <li>• Constants.FV_LANG_PORTUGUESE (0x0A)</li> <li>• Constants.FV_LANG_SPANISH (0x07)</li> <li>• Constants.FV_LANG_SWEDISH (0x11)</li> <li>• Constants.FV_LANG_SWISS_GERMAN (0x04)</li> <li>• Constants.FV_LANG_JAPANESE (0x12)</li> <li>• Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>• Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>• Constants.FV_LANG_KOREAN (0x15)</li> </ul>
FontSize	int	The font size (2 pt to 400 pt).
FontSizeChange	int	Amount by which to increase or decrease the font size.
FontVariation	int	Font variation (specifies an index into the array of font variations provided by the session property <code>FontVariationNames</code> ).
FontWeight	int	Font weight (specifies an index into the array of font weights provided by the session property <code>FontWeightNames</code> ).
KernX	int	Horizontal kern value for manual kerning expressed as a percentage of an em (metric – 100% to 1000%). a A positive value moves a character right and a negative value moves a character left.

Property name	Data Type	Description
KernY	int	Vertical kern value for manual kerning expressed as a percentage of an em (metric – 100% to 1000%). A positive value moves characters up and a negative value moves characters down.
Overline	int	1 if <code>Overline</code> is enabled. 0, otherwise.
PairKern	int	1 if <code>Pair Kern</code> is enabled. 0, otherwise.
Position	int	Specifies position relative to baseline of text. The value is one of: <ul style="list-style-type: none"> <li>• <code>Constants.FV_POS_NORM</code> (0): Normal</li> <li>• <code>Constants.FV_POS_SUPER</code> (1): Superscript</li> <li>• <code>Constants.FV_POS_SUB</code> (2): Subscript</li> </ul>
Stretch	int	Character stretch (set width) expressed as a percentage of normal stretch for the font (metric – 10% to 1000%).
StretchChange	int	Amount expressed as a percentage (metric – 10% to 1000%) by which to increase or decrease the character stretch.
Strikethrough	int	1 if <code>Strikethrough</code> is enabled. 0, otherwise.
Underlining	int	Type of underlining. The value is one of: <ul style="list-style-type: none"> <li>• <code>Constants.FV_CB_NO_UNDERLINE</code> (0)</li> <li>• <code>Constants.FV_CB_SINGLE_UNDERLINE</code> (1)</li> <li>• <code>Constants.FV_CB_DOUBLE_UNDERLINE</code> (2)</li> <li>• <code>Constants.FV_CB_NUMERIC_UNDERLINE</code> (3)</li> </ul>
BlockLines	int	The number of Widow/Orphan lines.
KeepWithNext	int	1 if <code>Keep With Next Paragraph</code> is enabled. 0 otherwise.
KeepWithPrev	int	1 if <code>Keep With Previous Paragraph</code> is enabled. 0 otherwise.
Placement	int	Denotes the Paragraph placement. The value is one of: <ul style="list-style-type: none"> <li>• <code>Constants.FV_PGF_SIDEBODY</code> (0)</li> <li>• <code>Constants.FV_PGF_SIDEHEAD_TOP</code> (1)</li> <li>• <code>Constants.FV_PGF_SIDEHEAD_FIRST_BASELINE</code> (2)</li> <li>• <code>Constants.FV_PGF_SIDEHEAD_LAST_BASELINE</code> (3)</li> <li>• <code>Constants.FV_PGF_RUN_IN</code> (4)</li> <li>• <code>Constants.FV_PGF_STRADDLE</code> (5)</li> <li>• <code>Constants.FV_PGF_STRADDLE_NORMAL_ONLY</code> (6)</li> </ul>
RunInSeparator	string	Denotes the string for Run-In Head Default Punctuation.

Property name	Data Type	Description
Start	int	Denotes the Vertical placement of a paragraph. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PGF_ANYWHERE ()</li> <li>• Constants.FV_PGF_TOP_OF_COL ()</li> <li>• Constants.FV_PGF_TOP_OF_PAGE ()</li> <li>• Constants.FV_PGF_TOP_OF_LEFT_PAGE ()</li> <li>• Constants.FV_PGF_TOP_OF_RIGHT_PAGE ()</li> </ul>
CellBottomMargin	int	Amount added to default bottom margin of table cell.
CellBottomMarginChange	int	Amount by which to increase or decrease the cell bottom margin.
CellBottomMarginFixed	int	1 if the cell bottom margin is fixed. 0 otherwise.
CellLeftMargin	int	Amount added to default left margin of table cell.
CellLeftMarginChange	int	Amount by which to increase or decrease the cell left margin.
CellLeftMarginFixed	int	1 if the cell left margin is fixed. 0 otherwise.
CellRightMargin	int	Amount added to default right margin of table cell.
CellRightMarginFixed	int	1 if the cell right margin is fixed. 0, otherwise.
CellTopMargin	int	Amount added to default top margin of table cell.
CellTopMarginChange	int	Amount by which to increase or decrease the cell top margin.
CellTopMarginFixed	int	1 if the cell top margin is fixed. 0, otherwise.
CellVAlignment	int	Denotes the vertical alignment of a paragraph when it is the first one in a cell. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PGF_V_ALIGN_TOP (0)</li> <li>• Constants.FV_PGF_V_ALIGN_MIDDLE (1)</li> <li>• Constants.FV_PGF_V_ALIGN_BOTTOM (2)</li> </ul>
BkColor	Color	Denotes the text background color. If this property is applied on a paragraph format, it denotes the background color of the whole paragraph; if it is applied on a character format, it denotes the background color of the text over which the character format is applied.
PgfBoxColor	Color	Denotes the background color of the paragraph box that encloses the paragraph text.

## FmtChangeList methods

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

## FmtRule

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

### FmtRule methods

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

## FmtRuleClause

Property name	Data Type	Description
ContextLabel	string	The context label for generated files. The label cannot contain white-space characters or any of these special characters: ( ) &   , * + ? < > % [ ] = ! ; : { } " <p>When a user displays the Set Up dialog box to set up a generated file, the label appears next to elements to which the rule clause applies.</p>
ElemPrefixSuffix	string	The text of the prefix or suffix. ElemPrefixSuffix specifies NULL if there is no prefix or suffix.
FmtChangeList	FmtChangeList	If the format rule clause specifies a format change list (RuleClauseType specifies Constants.FV_RC_CHANGE_LIST ()), FmtChangeList specifies the ID of the format change list (FO_FmtChangeList ID). To change the FmtChangeList property, use the NewFmtRuleObject () property.



Property name	Data Type	Description
FmtChangeListTag	string	Denotes the change list's tag, if the format rule clause specifies a change list (RuleClauseType specifies Constants.FV_RC_CHANGE_LIST_TAG (3)).
FmtRule	FmtRule	The ID of the format rule containing the format rule clause (FO_FmtRule ID).
FormatTag	string	The format tag if the format rule clause specifies one (RuleClauseType specifies Constants.FV_RC_TAG (0)). If IsTextRange is True, FormatTag specifies a character format tag; otherwise it specifies a paragraph tag, table tag, marker type, cross-reference format, or equation size.
IsTextRange	int	1 if the container element is formatted as a text range instead of a paragraph; 0 otherwise.
RuleClauseType	int	The type of rule clause. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_RC_TAG (0)</li> <li>Constants.FV_RC_SUB_FMTRUL (1)</li> <li>Constants.FV_RC_CHANGE_LIST (2)</li> <li>Constants.FV_RC_CHANGE_LIST_TAG (3)</li> </ul>
Specification	string	The format clause's context or level specification.
SpecificationForCSS	Strings	A list of CSS specifications that match the Specification for the current clause. For example, assume an element has a specification of F < (G   H). This property returns the following strings:  G > F > E and  H > F > E.
SubFmtRule	FmtRule	Denotes the the format rule's ID (FO_FmtRule ID), if the format rule clause contains a nested format rule (if RuleClauseType specifies Constants.FV_RC_SUB_FMTRULE (1)),

## FmtRuleClause methods

Delete, GetProps, NewFmtChangeList, NewSubFmtRule, ObjectValid, SetProps.

## Fn

Property name	Data Type	Description
ContentHeight	int	The distance between the top of the footnote and the baseline of the last line in the footnote.
Element	Element	Specifies the the ID of the element containing the footnote, if the footnote is in a FrameMaker document.
FirstPgf	Pgf	First paragraph in the footnote (FO_Pgf ID).
FnNum	int	Footnote number.

Property name	Data Type	Description
InTextFrame	<a href="#">TextFrame</a>	Text frame containing the footnote (FO_TextFrame ID).
InTextObj	<a href="#">SubCol</a>	Sub column that contains the footnote (FO_SubCol).
LastPgf	<a href="#">Pgf</a>	Last paragraph in the footnote (FO_Pgf ID).
NextFnInDoc	<a href="#">Fn</a>	Next footnote (FO_Fn ID) in the document.
NextFn	<a href="#">Fn</a>	Next footnote in the text frame (FO_Fn ID).
Overflowed	int	1 if the text in the footnote overflows; 0, otherwise.
PrevFn	<a href="#">Fn</a>	Previous footnote in the text frame (FO_Fn ID).
TextLoc	<a href="#">TextLoc</a>	Text location of the footnote symbol.
Unique	int	Footnote's unique ID.
FnAnchorString	string	This is the display string for a footnote anchor (generally a number string). This is a read-only property of a footnote and cannot be set by the user.

## Fn methods

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

## Font

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

## Font methods

[Font](#).

## Fonts

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

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

## Fonts methods

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

## Group

Property name	Data Type	Description
FirstGraphicInGroup	FObject	First object in the group.
LastGraphicInGroup	FObject	Last object in the group.
Angle	int	Angle of the object's rotation.
ArrowBaseAngle	int	Arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to False.
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Arrowhead tip angle in degrees.
ArrowType	int	Arrowhead style. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_ARROW_STICK (0x1)</li> <li>• Constants.FV_ARROW_HOLLOW (0x2)</li> <li>• Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Denotes the border width (0.015 points to 360 points).
Color	Color	The spot color (FO_Color ID).
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	The fill pattern (numbers between 0 and 15). The ESTK provides constants for the following fill patterns: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR(15)</li> </ul>
FrameParent	FObject	Frame containing the graphic object (FO_AFrame or FO_UnanchoredFrame ID).
GraphicCantBeSelected	int	1 if the graphic object cannot be selected; 0, otherwise.
GraphicsSelected	int	1 if the graphic object is selected; 0, otherwise.
GroupParent	Group	Group that the object is in (FO_Group ID).  <i>Note: Anchored and unanchored frames do not have this property.</i>

Property name	Data Type	Description
HeadArrow	int	1 if the line has an arrowhead; 0, otherwise.
Height	int	Height of the object (0.125 points to 3600 points).
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Type of line end. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Distance from the left side of the parent frame (-216 inches to 216 inches). If the graphic object is an anchored frame, the distance is calculated from the left side of the page frame. You cannot set LocX for anchored frames.
LocY	int	Distance from the top of the parent frame (-216 inches to 216 inches). If the graphic object is an anchored frame, the distance is calculated from the top of the page frame. You cannot set LocY for anchored frames.
NextGraphicInDoc	FObject	Next graphic object in the document.
NextGraphicInFrame	FObject	Next graphic object in the frame.
NextGraphicInGroup	FObject	Next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Next selected graphic object in document.
ObjectAttributes	Strings	A list of strings, each string expressing an attribute that is specified for an anchored frame in the <code>Object Properties &gt; Object Attributes</code> dialog box. Each string is in the form of <code>&lt;tag&gt;string_text&lt;/tag&gt;</code> . You can use the backslash to escape characters such as <code>\n</code> , <code>\r</code> , or <code>\t</code> for newline, carriage return, and tab, respectively. For a backslash character in the string text, type <code>\\</code> .
Overprint	int	Specifies the overprint settings for the object. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	The pen pattern (numbers between 0 and 7). The ESTK provides constants for the following values: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>

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

## Group methods

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

## HiddenPage

Property name	Data Type	Description
Name	string	Name of the hidden page.
PageFrame	<a href="#">UnanchoredFrame</a>	Denotes the page frame (FO_UnanchoredFrame ID)
PageHeight	int	Height of the hidden page.
PageWidth	int	Width of the hidden page.

## HiddenPage methods

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

## GraphicsFmt

Property name	Data Type	Description
UseColumnsAreBalanced	int	1 if terminal and underfilled columns in the flow are balanced.
UseSideHeadPlacement	int	Placement of side heads relative to columns in the text frame:  FV_SH_LEFT  FV_SH_RIGHT  FV_SH_INSIDE  FV_SH_OUTSIDE
UseRadius	MetricT	Radius of corner; 0 for a square corner
UseFlowsAutoConnect	Int	1 if Autoconnect is enabled.
UseFlowsPostScript	int	1 if flow is PostScript code.
UseBorderWidth	MetricT	Border width (0.015 pt to 360 pt).
UseFill	int	The fill pattern (numbers between 0 and 15). The FDK provides constants for several fill patterns:  FV_FILL_BLACK  FV_FILL_WHITE  FV_FILL_CLEAR
UsePen	int	The pen pattern (numbers between 0 and 15). The FDK provides constants for several pen patterns:  FV_FILL_BLACK  FV_FILL_WHITE  FV_FILL_CLEAR
UseRunaroundGap	MetricT	If the object is a runaround object, the width of the runaround gap.
UseTintPercent	MetricT	The tint percentage
UseOverprint	int	Specifies the overprint settings for the object:  FV_OVERPRINT  FV_KNOCKOUT  FV_FROMCOLOR
UseAngle	MetricT	Angle of the object's rotation.
UseLocX	MetricT	Distance from the left side of the parent frame.
UseLocY	MetricT	Distance from the top side of the parent frame
UseWidth	MetricT	Width of object
UseHeight	MetricT	Height of object.
UseRunaround	MetricT	If the object is a runaround object, the width of the runaround gap.

## Inset methods

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

## IdValuePair

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

## IdValuePair methods

## IdValuePairs

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

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

## Inset

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

Property name	Data Type	Description
InsetGfxName	String	Assigns a name to a graphic object. It will work only in case of inset objects that have an FLV, U3D or SWF facet. The name of the graphic should not contain any special characters or spaces.
InsetGfxPlayWindowInPdf	Bool	If this property is set, on publishing a document to PDF, the inset object that has facets FLV, U3D, or SWF will be activated in a new window in a PDF file. In PDF, the graphic objects are called annotation.
InsetFile	string	Platform-specific pathname if the inset is an external inset, or a null string (" ") if it is internal. The pathname can be document-relative.
InsetIsFixedSize	int	1 if scaling of bitmap file is inhibited. 0, otherwise.
InsetIsFlippedSideways	int	1 if inset is flipped about the vertical axis. 0, otherwise.
InsetJavaScriptAttached	Bool	Denotes whether or not Javascript is attached with the graphic object that has a U3D facet.
InsetJavaScriptFile	String	Attaches the given Javascript file to the graphic object that has a U3D facet. If the value of the file path is null, the Javascript attached to the inset is removed.
InsetMonikerFilePath	String	Provides the file path of the moniker of an inset object that has an OLE2 facet.
InsetSaveFacetToFile	Strings	Saves the given facet of an inset to a given file. This is set-only property and no get operations are possible. The input to the method contain the property name and list of strings. In the list of strings the, first strings is the facet name and the second string is the filename where the facet is to be saved.
InsetU3dAnimationList	Strings	Provides the list of "animations" defined in the U3D facet of an inset object.
InsetU3dPartList	Strings	Provides the list of "parts" defined in the U3D facet of an inset object.
InsetU3dViewList	Strings	Retrieves the list of "views" defined in the U3D facet of an inset object.
InsetUpdater	string	Not currently implemented.
PageNum	int	The page number of the PDF file to display in the inset, when importing a PDF file.  <i>Note: PDF page numbering begins with 0.</i>
InsetPosterFile	string	Sets the specified file as a poster on the Inset. This property is not supported for all the inset objects but supported only for multimedia objects and u3d object.  The property takes the poster file path as the value.
InsetGfxActiveInPdf	string	If this property is set, on publishing a document to PDF, the inset object (which has facets FLV, U3d, or swf) will be activated as soon as the page containing the graphic object is visible. In PDF the graphic objects are called annotation.
InsetJavaScriptFile	string	Attaches the given Javascript file to the graphic object that has a U3d facet. If the value of the file path is null, the Javascript attached to the inset is removed.



Property name	Data Type	Description
InsetU3dAnimationList	string	A read-only property. Provides the list of "animations" defined on the U3d facet of a inset object.
InsetU3dPartList	string	Provides the list of "parts" defined in the U3d facet of a inset object.
InsetU3dViewList	string	A read-only property. Retrieves the list of "views" defined in the U3d facet of a inset object.
Angle	int	Angle of the object's rotation.
ArrowBaseAngle	int	Arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to False.
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Arrowhead tip angle in degrees.
ArrowType	int	Arrowhead style. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_ARROW_STICK (0x1)</li> <li>• Constants.FV_ARROW_HOLLOW (0x2)</li> <li>• Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Denotes the border width (0.015 points to 360 points).
Color	Color	The spot color (FO_Color ID)
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements
Fill	int	The fill pattern (numbers between 0 and 15). The ESTK provides constants for the following fill patterns: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR(15)</li> </ul>
FrameParent	FObject	Frame containing the graphic object (FO_AFrame or FO_UnanchoredFrame ID).
GraphicCantBeSelected	int	1 if the graphic object cannot be selected; 0, otherwise.
GraphicsSelected	int	1 if the graphic object is selected; 0, otherwise.
GroupParent	Group	Group that the object is in (FO_Group ID).  <i>Note: Anchored and unanchored frames do not have this property.</i>

Property name	Data Type	Description
HeadArrow	int	1 if the line has an arrowhead; 0, otherwise.
Height	int	Height of the object (0.125 points to 3600 points).
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Type of line end. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Distance from the left side of the parent frame (-216 inches to 216 inches). If the graphic object is an anchored frame, the distance is calculated from the left side of the page frame. You cannot set LocX for anchored frames.
LocY	int	Distance from the top of the parent frame (-216 inches to 216 inches). If the graphic object is an anchored frame, the distance is calculated from the top of the page frame. You cannot set LocY for anchored frames.
NextGraphicInDoc	FObject	Next graphic object in the document.
NextGraphicInFrame	FObject	Next graphic object in the frame.
NextGraphicInGroup	FObject	Next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Next selected graphic object in document.
ObjectAttributes	Strings	A list of strings, each string expressing an attribute that is specified for an anchored frame in the <code>Object Properties &gt; Object Attributes</code> dialog box. Each string is in the form of <code>&lt;tag&gt;string_text&lt;/tag&gt;</code> . You can use the backslash to escape characters such as <code>\n</code> , <code>\r</code> , or <code>\t</code> for newline, carriage return, and tab, respectively. For a backslash character in the string text, type <code>\\</code> .
Overprint	int	Specifies the overprint settings for the object. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	The pen pattern (numbers between 0 and 7). The ESTK provides constants for the following values: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>

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

## Inset methods

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

## Ints

An Array ofobjects with integer indexing and a *length* property.

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

## Ints methods

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

## KeyCatalog

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

## Line

Property name	Data Type	Description
NumPoints	int	Specifies the number of vertices. The default is 2 (the line's start point and end point).
Points	Points	Array of x-y coordinate pairs that specify the line's vertices. The default coordinate pairs are for the line's start point and end point.
Angle	int	Angle of the line's rotation.
ArrowBaseAngle	int	Arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to False.
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Arrowhead tip angle in degrees.

Property name	Data Type	Description
ArrowType	int	Arrowhead style. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Denotes the border width (0.015 points to 360 points).
Color	Color	The spot color (FO_Color ID)
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements
Fill	int	The fill pattern (numbers between 0 and 15). The ESTK provides constants for the following fill patterns: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR(15)</li> </ul>
FrameParent	FObject	Frame containing the graphic object (FO_AFrame or FO_UnanchoredFrame ID).
GraphicCantBeSelected	int	1 if the graphic object cannot be selected; 0, otherwise.
GraphicIsSelected	int	1 if the graphic object is selected; 0, otherwise.
GroupParent	Group	Group that the line is in (FO_Group ID).
HeadArrow	int	1 if the line has an arrowhead; 0, otherwise.
Height	int	Height of the object (0.125 points to 3600 points).
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Type of line end. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Distance from the left side of the parent frame (-216 inches to 216 inches). If the graphic object is an anchored frame, the distance is calculated from the left side of the page frame. You cannot set LocX for anchored frames.

Property name	Data Type	Description
LocY	int	Distance from the top of the parent frame (-216 inches to 216 inches).If the graphic object is an anchored frame, the distance is calculated from the top of the page frame. You cannot set LocY for anchored frames.
NextGraphicInDoc	FMOBJECT	Next graphic object in the document.
NextGraphicInFrame	FMOBJECT	Next graphic object in the frame.
NextGraphicInGroup	FMOBJECT	Next graphic object in the group.
NextSelectedGraphicInDoc	FMOBJECT	Next selected graphic object in document.
ObjectAttributes	Strings	A list of strings, each string expressing an attribute that is specified for an anchored frame in the <b>Object Properties &gt; Object Attributes</b> dialog box.Each string is in the form of <tag>string_text</tag>. You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	The pen pattern (numbers between 0 and 7). The ESTK provides constants for the following values: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FMOBJECT	Previous graphic object in the frame.
PrevGraphicInGroup	FMOBJECT	Previous graphic object in the group
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_TR_NONE (0x01)</li> <li>• Constants.FV_TR_CONTOUR (0x02)</li> <li>• Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the the width of the runaround gapfor a runaround object.
TailArrow	int	1 if arrowhead is present at the beginning of the line. 0, otherwise.
TintPercent	int	The tint percentage.
Unique	int	Unique ID of the object.
Width	int	Width of the object (0.125 points to 3600 points).

## Line methods

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

## Marker

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

## Marker methods

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

## MarkerType

Property name	Data Type	Description
NextMarkerTypeInDoc	<a href="#">MarkerType</a>	Denotes the next marker type (FO_MarkerType ID).
Name	string	The name of this marker type, as it appears in the user interface.
InvariantName	string	An internal name for the marker type. By default, this is the same as Name. However, this can differ from Name if the user interface is in another language.
Public	bool	True if the marker type should appear in the user interface. The default is True.
Transient	bool	True if markers of this type should not be saved to files. The default is False.
Required	bool	True if the marker type is required by FrameMaker. The default is False.

## MarkerType methods

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

## MasterPage

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

### MasterPage methods

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

## Math

Property name	Data Type	Description
BasePointX	int	Horizontal placement of text line base point relative to the left side of the frame.
BasePointY	int	Vertical placement of text line base point relative to top of the frame.
MathFullForm	string	String representing the mathematical expression.
MathSize	int	The equation size. The value is one of : <ul style="list-style-type: none"> <li>• Constants.FV_MATH_MEDIUM (0)</li> <li>• Constants.FV_MATH_SMALL (1)</li> <li>• Constants.FV_MATH_LARGE (2)</li> </ul>
TextLineType	int	Type of text line. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_TEXTLINE_LEFT (0)</li> <li>• Constants.FV_TEXTLINE_RIGHT (1)</li> <li>• Constants.FV_TEXTLINE_CENTER (2)</li> <li>• Constants.FV_TEXTLINE_MATH (3)</li> </ul>
Angle	int	Specifies the angle of the object's rotation.
ArrowBaseAngle	int	Indicates the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).



Property name	Data Type	Description
ArrowScaleHead	int	Set to 1 if the arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to False.
ArrowTipAngle	int	Specifies the arrowhead tip angle in degrees.
ArrowType	int	Denotes the Arrowhead style. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_ARROW_STICK (0x01)</li> <li>• Constants.FV_ARROW_HOLLOW (0x02)</li> <li>• Constants.FV_ARROW_FILLED (0x03)</li> </ul>
BorderWidth	int	Specifies the border width in points. The permissible range is 0.015 point to 360 point.
Color	Color	Indicates the spot color.
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a Metrics object. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern. The patterns are denoted by numbers between 0 and 15.  The following constants are provided for the fill patterns 0, 7 and 15: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Specifies the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Else, set to 0.
GraphicsSelected	int	Set to 1 if the graphic object is selected, else set to 0.
GroupParent	Group	Indicates the group to which the object belongs. Anchored and unanchored frames do not have this property.
HeadArrow	int	Set to 1 if line has an arrowhead, set to 0 otherwise.
Height	int	Specifies the height of the object in points. The permissible range is from 0.125 pt to 3600 pt.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.

Property name	Data Type	Description
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Indicates the type of line end. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent, in inches. The permissible range is from -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set the LocX property for anchored frames.
LocY	int	Specifies the distance of the object from the top of the parent frame, in inches. The permissible range is from -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You CANNOT set the LocY property for anchored frames.
NextGraphicInDoc	FObject	Indicates the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Indicates the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Specifies the next selected graphic object in the document.
ObjectAttributes	Strings	Contains a list of strings, each string expressing an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Indicates the pen pattern, denoted by numbers between 0 and 7.  Constants are provided for some of the fill patterns: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>

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

## Math methods

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

## MathML

Property name	Data Type	Description
MathMLDpi	int	Specifies the display Dpi for the MathML object.
MathMLIsFixedSize	int	1 if scaling of bitmap file is inhibited. 0, otherwise.
MathMLIsInverted	int	1 if image is flipped about the horizontal axis. 0, otherwise.
MathMLIsFlippedSideways	int	1 if image is flipped about the vertical axis. 0, otherwise.
MathMLData	strings	Specifies the XML data of the MathML object as a string array.
MathMLComposeDpi	int	Specifies the compose Dpi for the MathML object
MathMLFontSize	int	Specifies the font size for the MathML object
MathMLXmlData	string	Specifies the XML data of the MathML object as a string.
Angle	int	Specifies the angle of the object's rotation.
ArrowBaseAngle	int	Indicates the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).

Property name	Data Type	Description
ArrowScaleHead	int	Set to 1 if the arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowScaleFactor	int	Specifies the factor by which arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is set to False.
ArrowTipAngle	int	Specifies the arrowhead tip angle in degrees.
ArrowType	int	Denotes the Arrowhead style. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_ARROW_STICK (0x01)</li> <li>• Constants.FV_ARROW_HOLLOW (0x02)</li> <li>• Constants.FV_ARROW_FILLED (0x03)</li> </ul>
BorderWidth	int	Specifies the border width in points. The permissible range is 0.015 point to 360 point.
Color	Color	Indicates the spot color.
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a Metrics object. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern. The patterns are denoted by numbers between 0 and 15.  The following constants are provided for the fill patterns 0, 7 and 15: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Specifies the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Else, set to 0.
GraphicsSelected	int	Set to 1 if the graphic object is selected, else set to 0.
GroupParent	Group	Indicates the group to which the object belongs. Anchored and unanchored frames do not have this property.
HeadArrow	int	Set to 1 if line has an arrowhead, set to 0 otherwise.
Height	int	Specifies the height of the object in points. The permissible range is from 0.125 pt to 3600 pt.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.

Property name	Data Type	Description
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Indicates the type of line end. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent, in inches. The permissible range is from -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set the LocX property for anchored frames.
LocY	int	Specifies the distance of the object from the top of the parent frame, in inches. The permissible range is from -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You CANNOT set the LocY property for anchored frames.
NextGraphicInDoc	FObject	Indicates the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Indicates the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Specifies the next selected graphic object in the document.
ObjectAttributes	Strings	Contains a list of strings, each string expressing an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Indicates the pen pattern, denoted by numbers between 0 and 7.  Constants are provided for some of the fill patterns: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>

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

## MathML methods

[Delete](#), [GetProps](#), [SetProps](#) on page 653, [ObjectValid](#) on page 653.

## Menu

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

Property name	Data Type	Description
NextMenuItemInMenu	FObject	The next menu item, menu, or separator in the menu.
NextMenuItemInSession	FObject	The next menu item, menu, or separator in the list of menu items, menus, and separators in the session.
PrevMenuItemInMenu	FObject	The previous menu item, menu, or separator in the list of menu items, menus, and separators in the session.

## Menu methods

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

## MenuItemSeparator

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

## MenuItemSeparator methods

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

## Metrics

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

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

## Metrics methods

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

## Pgf

Property name	Data Type	Description
MinJRomSpace	int	Minimum Asian-Roman space.
OptJRomSpace	int	Optimum Asian-Roman space.
MaxJRomSpace	int	Maximum Asian-Roman space.
MinJLetSpace	int	Minimum Asian letter space.
OptJLetSpace	int	Optimum Asian letter space.
MaxJLetSpace	int	Maximum Asian letter space.
YakumonoType	int	The Yakumono rules to handle punctuation characters. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_FLOATING_YAKUMONO (0x00)</li> <li>Constants.FV_MONOSPACE_YAKUMONO (0x01)</li> <li>Constants.FV_FIXED_YAKUMONO (0x02)</li> </ul>
AutoNumChar	string	Character format for the automatic numbering string specified by <code>AutoNumString</code> ; " " if the default character format is used
AutoNumString	string	Autonumber format string (for example, <n>.<n+>)
NumAtEnd	int	1 if numbering position is End of Paragraph; 0 if it is Beginning of Paragraph.
PgflsAutoNum	int	1 if autonumbering is enabled.
PgfNumber	string	The formatted string representation of the paragraph number; for example, 1.2 for a paragraph whose <code>AutoNumString</code> property is set to <n>.<n+>
Capitalization	int	Type of capitalization to use. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAPITAL_CASE_NORM (0)</li> <li>Constants.FV_CAPITAL_CASE_SMALL (1)</li> <li>Constants.FV_CAPITAL_CASE_LOWER (2)</li> <li>Constants.FV_CAPITAL_CASE_UPPER (3)</li> </ul>
ChangeBar	int	1 if Change Bars are on.
Color	Color	Spot color (FO_Color ID).
CombinedFontFamily	CombinedFontDefn	Combined font definition (FO_CombinedFontDefn).
FontAngle	int	Font angle (specifies an index into the array of font angles provided by the session property <code>FontAngleNames</code> ).
FontEncodingName	string	The font's encoding.
FontFamily	int	Font family (specifies an index into the array of font families provided by the session property <code>FontFamilyNames</code> ).
FontPlatformName	string	Name that uniquely identifies a font on a specific platform.



Property name	Data Type	Description
FontPostScriptName	string	Name given to a font when it is sent to a PostScript printer.
FontSize	int	Font size (2 pt to 400 pt).
FontVariation	int	Font variation (specifies an index into the array of font variations provided by the session property <code>FontVariationNames</code> ).
FontWeight	int	Font weight (specifies an index into the array of font weights provided by the session property <code>FontWeightNames</code> ).
KernX	int	Horizontal kern value for manual kerning expressed as a percentage of an em (metric –100% to 1000%). A positive value moves a character right and a negative value moves a character left.
KernY	int	Vertical kern value for manual kerning expressed as a percentage of an em (metric –100% to 1000%). A positive value moves characters up and a negative value moves characters down.
Overline	int	1 if Overline is enabled.
PairKern	int	1 if Pair Kern is enabled.
Position	int	Specifies position relative to baseline of text. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_POS_NORM (0): Normal</li> <li>• Constants.FV_POS_SUPER (1): Superscript</li> <li>• Constants.FV_POS_SUB (2): Subscript</li> </ul>
Stretch	int	Character stretch (set width) expressed as a percentage of normal stretch for the font (metric –10% to 1000%).
Strikethrough	int	1 if Strikethrough is enabled.
Underlining	int	Type of underlining. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_CB_NO_UNDERLINE (0)</li> <li>• Constants.FV_CB_SINGLE_UNDERLINE (1)</li> <li>• Constants.FV_CB_DOUBLE_UNDERLINE (2)</li> <li>• Constants.FV_CB_NUMERIC_UNDERLINE (3)</li> </ul>
CellBottomMargin	int	Amount added to default bottom margin of table cell.
CellLeftMargin	int	Amount added to default left margin of table cell.

Property name	Data Type	Description
CellMarginsFixed	int	<p>Specifies which cell margins are fixed. The following values can be ORed into it:</p> <ul style="list-style-type: none"> <li>• Constants.FV_PGF_FIXED_L_MARGIN (0x01): the left margin is fixed.</li> <li>• Constants.FV_PGF_FIXED_B_MARGIN (0x02): the bottom margin is fixed.</li> <li>• Constants.FV_PGF_FIXED_R_MARGIN (0x04): the right margin is fixed.</li> <li>• Constants.FV_PGF_FIXED_T_MARGIN (0x08): the top margin is fixed.</li> </ul> <p>If the margin for a cell is fixed, the margin property specifies the absolute value of the cell margin. For example, if <code>Constants.FV_PGF_FIXED_B_MARGIN (0x02)</code> is set, <code>CellBottomMargin</code> specifies the absolute value of the cell's bottom margin, overriding the cell margin specified by the table format.</p> <p>If <code>Constants.FV_PGF_FIXED_B_MARGIN (0x02)</code> is not set, <code>CellBottomMargin</code> is added to the margin specified by the table format.</p>
CellRightMargin	int	Amount added to default right margin of table cell.
CellTopMargin	int	Amount added to default top margin of table cell.
CellVAlignment	int	<p>Vertical alignment of a paragraph when it is the first one in a cell. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_PGF_V_ALIGN_TOP (0)</li> <li>• Constants.FV_PGF_V_ALIGN_MIDDLE (1)</li> <li>• Constants.FV_PGF_V_ALIGN_BOTTOM (2)</li> </ul>
Locked	int	1 if the paragraph is part of a text inset that retains formatting information from the source document. The paragraph is not affected by global formatting performed on the document.
FormatOverride	int	1 if the paragraph contains a paragraph format override.
AdjHyphens	int	Number of allowable adjacent hyphens.
Hyphenate	int	1 if Automatic Hyphenation is enabled.
HyphMinPrefix	int	Minimum number of letters that must precede hyphen.
HyphMinSuffix	int	Minimum number of letters that must follow a hyphen.
HyphMinWord	int	Minimum length of a hyphenated word.

Property name	Data Type	Description
Language	int	<p>Hyphenation and spell-checking language to use. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li>• Constants.FV_LANG_ENGLISH (0x01)</li> <li>• Constants.FV_LANG_BRITISH (0x02)</li> <li>• Constants.FV_LANG_GERMAN (0x03)</li> <li>• Constants.FV_LANG_SWISS_GERMAN (0x04)</li> <li>• Constants.FV_LANG_FRENCH (0x05)</li> <li>• Constants.FV_LANG_CANADIAN_FRENCH (0x06)</li> <li>• Constants.FV_LANG_SPANISH (0x07)</li> <li>• Constants.FV_LANG_CATALAN (0x08)</li> <li>• Constants.FV_LANG_ITALIAN (0x09)</li> <li>• Constants.FV_LANG_FINNISH (0x10)</li> <li>• Constants.FV_LANG_SWEDISH (0x11)</li> <li>• Constants.FV_LANG_JAPANESE (0x12)</li> <li>• Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>• Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>• Constants.FV_LANG_KOREAN (0x15)</li> <li>• Constants.FV_LANG_PORTUGUESE (0x0A)</li> <li>• Constants.FV_LANG_BRAZILIAN (0x0B)</li> <li>• Constants.FV_LANG_DANISH (0x0C)</li> <li>• Constants.FV_LANG_DUTCH (0x0D)</li> <li>• Constants.FV_LANG_NORWEGIAN (0x0E)</li> <li>• Constants.FV_LANG_NYNORSK (0x0F)</li> </ul>
PgfSpellChecked	int	1 if paragraph has been spell-checked
Unique	int	The paragraph's user ID.
FirstIndent	int	First-line left margin, measured from left side of current text column (0 cm to 100 cm).
LeftIndent	int	Left margin, measured from left side of current text column (0 cm to 100 cm).
RightIndent	int	Right margin, measured from right side of current text column.
Leading	int	Space below each line in a paragraph.

Property name	Data Type	Description
LineSpacing	int	Space between lines in a paragraph measured from baseline to baseline. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PGF_FIXED (0x00): default font size</li> <li>• Constants.FV_PGF_PROPORTIONAL (0x01): largest font in line</li> <li>• Constants.FV_PGF_FLOATING (0x02): largest ascender in line</li> </ul>
BlockLines	int	The number of Widow/Orphan lines.
KeepWithNext	int	1 if Keep With Next Paragraph is enabled.
KeepWithPrev	int	1 if Keep With Previous Paragraph is enabled.
PgfAlignment	int	Horizontal alignment of paragraph. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PGF_LEFT (0x01)</li> <li>• Constants.FV_PGF_RIGHT (0x02)</li> <li>• Constants.FV_PGF_CENTER (0x03)</li> <li>• Constants.FV_PGF_JUSTIFIED (0x04)</li> </ul>
Placement	int	Paragraph placement. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PGF_SIDEBODY (0)</li> <li>• Constants.FV_PGF_SIDEHEAD_TOP (1)</li> <li>• Constants.FV_PGF_SIDEHEAD_FIRST_BASELINE (2)</li> <li>• Constants.FV_PGF_SIDEHEAD_LAST_BASELINE (3)</li> <li>• Constants.FV_PGF_RUN_IN (4)</li> <li>• Constants.FV_PGF_STRADDLE (5)</li> <li>• Constants.FV_PGF_STRADDLE_NORMAL_ONLY (6)</li> </ul>
RunInSeparator	string	String for Run-In Head Default Punctuation.
SpaceAbove	int	Space above paragraph.
SpaceBelow	int	Space below paragraph.
Start	int	Vertical placement of paragraph. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PGF_ANYWHERE (0x00)</li> <li>• Constants.FV_PGF_TOP_OF_COL (0x01)</li> <li>• Constants.FV_PGF_TOP_OF_PAGE (0x02)</li> <li>• Constants.FV_PGF_TOP_OF_LEFT_PAGE (0x03)</li> <li>• Constants.FV_PGF_TOP_OF_RIGHT_PAGE (0x04)</li> </ul>
InTextFrame	<a href="#">TextFrame</a>	Text frame containing the paragraph (FO_TextFrame ID).
InTextObj	FMOobject	Subcolumn, footnote, or table cell the paragraph begins in (FO_SubCol, FO_Fn, or FO_Cell ID).

Property name	Data Type	Description
NextPgflnDoc	<a href="#">PgflnDoc</a>	Next paragraph in the document (FO_PgflnDoc ID).
NextPgflnFlow	<a href="#">PgflnFlow</a>	Next paragraph in the flow (FO_PgflnFlow ID).
PrevPgflnFlow	<a href="#">PgflnFlow</a>	Previous paragraph in the flow (FO_PgflnFlow ID).
BottomSeparator	string	Name of frame to put below paragraph.
TopSeparator	string	Name of frame to put above paragraph.
NumTabs	int	Number of tabs in the paragraph..
Tabs	<a href="#">Tabs</a>	Array of tab descriptions that specify the positions and types of tab stops.
Name	string	Name of paragraph format.
NextTag	string	Tag for new next paragraph.
UseNextTag	int	1 if Next Paragraph Tag is enabled.
LetterSpace	int	1 if Word Spacing is enabled.
MaxSpace	int	Maximum word spacing (percentage of an em space in current font).
MinSpace	int	Minimum word spacing (percentage of an em space in current font).
OptSpace	int	Optimum word spacing.
BkColor	<a href="#">Color</a>	Denotes the text background color. If this property is applied on a paragraph format, it denotes the background color of the whole paragraph; if it is applied on a character format, it denotes the background color of the text over which the character format is applied.
PgflnBoxColor	<a href="#">Color</a>	Denotes the background color of the paragraph box that encloses the paragraph text.

## Pgfln methods

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

## PgfFmt

Property name	Data Type	Description
AcrobatLevel	int	Retained in Version 6.0 or later for backward compatibility. Use PDFStructureLevel instead.
PgfMarkedForNamedDestination	int	If 1, this paragraph will have a corresponding Named Destination in the generated PDF.
PDFStructureLevel	int	The PDF structure level of paragraphs with the current format. This property is used when the PDFStructure property is True for the document, and FrameMaker generates PDF data. The value for this property can be between 0 and 100, where greater values are deeper in the hierarchy. If PDFStructureLevel is 0, FrameMaker does not include paragraphs of this format in the PDF structure.
MinJRomSpace	int	Minimum Asian-Roman space.
OptJRomSpace	int	Optimum Asian-Roman space.
MaxJRomSpace	int	Maximum Asian-Roman space.
MinJLetSpace	int	Minimum Asian letter space.
OptJLetSpace	int	Optimum Asian letter space.
MaxJLetSpace	int	Maximum Asian letter space.
YakumonoType	int	The Yakumono rules to handle punctuation characters. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_FLOATING_YAKUMONO (0x00)</li> <li>Constants.FV_MONOSPACE_YAKUMONO (0x01)</li> <li>Constants.FV_FIXED_YAKUMONO (0x02)</li> </ul>
AutoNumChar	string	Character format for the automatic numbering string specified by AutoNumString; " " if the default character format is used
AutoNumString	string	Autonumber format string (for example, <n>.<n+>)
NumAtEnd	int	1 if numbering position is End of Paragraph; 0 if it is Beginning of Paragraph.
PgflsAutoNum	int	1 if autonumbering is enabled.
Capitalization	int	Type of capitalization to use. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_CAPITAL_CASE_NORM (0)</li> <li>Constants.FV_CAPITAL_CASE_SMALL (1)</li> <li>Constants.FV_CAPITAL_CASE_LOWER (2)</li> <li>Constants.FV_CAPITAL_CASE_UPPER (3)</li> </ul>
ChangeBar	int	1 if Change Bars are on.
Color	Color	Spot color (FO_Color ID).
CombinedFontFamily	CombinedFontDefn	Combined font definition (FO_CombinedFontDefn).

Property name	Data Type	Description
FontAngle	int	Font angle (specifies an index into the array of font angles provided by the session property <code>FontAngleNames</code> ).
FontEncodingName	string	The font's encoding.
FontFamily	int	Font family (specifies an index into the array of font families provided by the session property <code>FontFamilyNames</code> ).
FontPlatformName	string	Name that uniquely identifies a font on a specific platform.
FontPostScriptName	string	Name given to a font when it is sent to a PostScript printer.
FontSize	int	Font size (2 pt to 400 pt).
FontVariation	int	Font variation (specifies an index into the array of font variations provided by the session property <code>FontVariationNames</code> ).
FontWeight	int	Font weight (specifies an index into the array of font weights provided by the session property <code>FontWeightNames</code> ).
KernX	int	Horizontal kern value for manual kerning expressed as a percentage of an em (metric –100% to 1000%). A positive value moves a character right and a negative value moves a character left.
KernY	int	Vertical kern value for manual kerning expressed as a percentage of an em (metric –100% to 1000%). A positive value moves characters up and a negative value moves characters down.
Overline	int	1 if Overline is enabled.
PairKern	int	1 if Pair Kern is enabled.
Position	int	Specifies position relative to baseline of text. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_POS_NORM (0): Normal</li> <li>• Constants.FV_POS_SUPER (1): Superscript</li> <li>• Constants.FV_POS_SUB (2): Subscript</li> </ul>
Stretch	int	Character stretch (set width) expressed as a percentage of normal stretch for the font (metric –10% to 1000%).
Strikethrough	int	1 if Strikethrough is enabled.
Underlining	int	Type of underlining. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_CB_NO_UNDERLINE (0)</li> <li>• Constants.FV_CB_SINGLE_UNDERLINE (1)</li> <li>• Constants.FV_CB_DOUBLE_UNDERLINE (2)</li> <li>• Constants.FV_CB_NUMERIC_UNDERLINE (3)</li> </ul>
AdjHyphens	int	Number of allowable adjacent hyphens
Hyphenate	int	1 if Automatic Hyphenation is enabled.

Property name	Data Type	Description
HyphMinPrefix	int	Minimum number of letters that must precede hyphen.
HyphMinSuffix	int	Minimum number of letters that must follow a hyphen.
HyphMinWord	int	Minimum length of a hyphenated word.
Language	int	<p>Hyphenation and spell-checking language to use. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li>• Constants.FV_LANG_ENGLISH (0x01)</li> <li>• Constants.FV_LANG_BRITISH (0x02)</li> <li>• Constants.FV_LANG_GERMAN (0x03)</li> <li>• Constants.FV_LANG_SWISS_GERMAN (0x04)</li> <li>• Constants.FV_LANG_FRENCH (0x05)</li> <li>• Constants.FV_LANG_CANADIAN_FRENCH (0x06)</li> <li>• Constants.FV_LANG_SPANISH (0x07)</li> <li>• Constants.FV_LANG_CATALAN (0x08)</li> <li>• Constants.FV_LANG_ITALIAN (0x09)</li> <li>• Constants.FV_LANG_FINNISH (0x10)</li> <li>• Constants.FV_LANG_SWEDISH (0x11)</li> <li>• Constants.FV_LANG_JAPANESE (0x12)</li> <li>• Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>• Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>• Constants.FV_LANG_KOREAN (0x15)</li> <li>• Constants.FV_LANG_PORTUGUESE (0x0A)</li> <li>• Constants.FV_LANG_BRAZILIAN (0x0B)</li> <li>• Constants.FV_LANG_DANISH (0x0C)</li> <li>• Constants.FV_LANG_DUTCH (0x0D)</li> <li>• Constants.FV_LANG_NORWEGIAN (0x0E)</li> <li>• Constants.FV_LANG_NYNORSK (0x0F)</li> </ul>
FirstIndent	int	First-line left margin, measured from left side of current text column (0 cm to 100 cm).
LeftIndent	int	Left margin, measured from left side of current text column (0 cm to 100 cm).
RightIndent	int	Right margin, measured from right side of current text column.
Leading	int	Space below each line in a paragraph.



Property name	Data Type	Description
LineSpacing	int	Space between lines in a paragraph measured from baseline to baseline. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_FIXED (0x00): default font size</li> <li>Constants.FV_PGF_PROPORTIONAL (0x01): largest font in line</li> <li>Constants.FV_PGF_FLOATING (0x02): largest ascender in line</li> </ul>
NextPgInDoc	Pgf	Next paragraph in the document (FO_Pgf ID).
BlockLines	int	The number of Widow/Orphan lines.
KeepWithNext	int	1 if Keep With Next Paragraph is enabled.
KeepWithPrev	int	1 if Keep With Previous Paragraph is enabled.
PgfAlignment	int	Horizontal alignment of paragraph. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_LEFT (0x01)</li> <li>Constants.FV_PGF_RIGHT (0x02)</li> <li>Constants.FV_PGF_CENTER (0x03)</li> <li>Constants.FV_PGF_JUSTIFIED (0x04)</li> </ul>
Placement	int	Paragraph placement. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_SIDEBODY (0)</li> <li>Constants.FV_PGF_SIDEHEAD_TOP (1)</li> <li>Constants.FV_PGF_SIDEHEAD_FIRST_BASELINE (2)</li> <li>Constants.FV_PGF_SIDEHEAD_LAST_BASELINE (3)</li> <li>Constants.FV_PGF_RUN_IN (4)</li> <li>Constants.FV_PGF_STRADDLE (5)</li> <li>Constants.FV_PGF_STRADDLE_NORMAL_ONLY (6)</li> </ul>
RunInSeparator	string	String for Run-In Head Default Punctuation.
SpaceAbove	int	Space above paragraph.
SpaceBelow	int	Space below paragraph.
Start	int	Vertical placement of paragraph. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_PGF_ANYWHERE (0x00)</li> <li>Constants.FV_PGF_TOP_OF_COL (0x01)</li> <li>Constants.FV_PGF_TOP_OF_PAGE (0x02)</li> <li>Constants.FV_PGF_TOP_OF_LEFT_PAGE (0x03)</li> <li>Constants.FV_PGF_TOP_OF_RIGHT_PAGE (0x04)</li> </ul>
BottomSeparator	string	Name of frame to put below paragraph.
TopSeparator	string	Name of frame to put above paragraph.

Property name	Data Type	Description
BkColor	Color	Denotes the text background color. If this property is applied on a paragraph format, it denotes the background color of the whole paragraph; if it is applied on a character format, it denotes the background color of the text over which the character format is applied.
PgfBoxColor	Color	Denotes the background color of the paragraph box that encloses the paragraph text.
UseBkColor	bool	The UseBkColor is a Boolean flag which indicates whether the BkColor property is set or not. A value of 1 indicates the text has some background color. A value of 0 indicates a transparent background.  <i>Note: When the UseBkColor property is set to 0, the BkColor value is not used.</i>
CellBottomMargin	int	Amount added to default bottom margin of table cell.
CellLeftMargin	int	Amount added to default left margin of table cell.
CellMarginsFixed	int	Specifies which cell margins are fixed. The following values can be ORed into it: <ul style="list-style-type: none"> <li>• Constants.FV_PGF_FIXED_L_MARGIN (0x01): the left margin is fixed.</li> <li>• Constants.FV_PGF_FIXED_B_MARGIN (0x02): the bottom margin is fixed.</li> <li>• Constants.FV_PGF_FIXED_R_MARGIN (0x04): the right margin is fixed.</li> <li>• Constants.FV_PGF_FIXED_T_MARGIN (0x08): the top margin is fixed.</li> </ul> <p>If the margin for a cell is fixed, the margin property specifies the absolute value of the cell margin. For example, if Constants.FV_PGF_FIXED_B_MARGIN (0x02) is set, CellBottomMargin specifies the absolute value of the cell's bottom margin, overriding the cell margin specified by the table format.</p> <p>If Constants.FV_PGF_FIXED_B_MARGIN (0x02) is not set, CellBottomMargin is added to the margin specified by the table format.</p>
CellRightMargin	int	Amount added to default right margin of table cell.
CellTopMargin	int	Amount added to default top margin of table cell.
CellVAlignment	int	Vertical alignment of a paragraph when it is the first one in a cell. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_PGF_V_ALIGN_TOP (0)</li> <li>• Constants.FV_PGF_V_ALIGN_MIDDLE (1)</li> <li>• Constants.FV_PGF_V_ALIGN_BOTTOM (2)</li> </ul>
NumTabs	int	Number of tabs in the paragraph..
Tabs	Tabs	Array of tab descriptions that specify the positions and types of tab stops.
Name	string	Name of paragraph format.

Property name	Data Type	Description
NextTag	string	Tag for new next paragraph.
UseNextTag	int	1 if Next Paragraph Tag is enabled.
LetterSpace	int	1 if Word Spacing is enabled.
MaxSpace	int	Maximum word spacing (percentage of an em space in current font).
MinSpace	int	Minimum word spacing (percentage of an em space in current font).
OptSpace	int	Optimum word spacing.

## PgFFmt methods

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

## Point

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

## Point methods

[Point](#).

## Points

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

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

## Points methods

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

# Polygon

Property name	Data Type	Description
NumPoints	int	Number of polygon vertices.
Points	Points	Array of x-y coordinate pairs that specify the polygon's vertices.
PolylsBezier	int	1 if polygon is smoothed.
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is False.
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	Color	Denotes the spot color.
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.

Property name	Data Type	Description
GroupParent	Group	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FMOBJECT	Denotes the next graphic object in the document.
NextGraphicInFrame	FMOBJECT	Denotes the next graphic object in the frame.
NextGraphicInGroup	FMOBJECT	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FMOBJECT	Denotes the next selected graphic object in the document.
ObjectAttributes	Strings	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.

Property name	Data Type	Description
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_TR_NONE (0x01)</li> <li>• Constants.FV_TR_CONTOUR (0x02)</li> <li>• Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Indicates the graphic object's internal unique number.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## Polygon methods

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

# Polyline

Property name	Data Type	Description
NumPoints	int	Number of polygon vertices.
Points	Points	Array of x-y coordinate pairs that specify the polygon's vertices.
PolylsBezier	int	1 if polygon is smoothed.
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is False.
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	Color	Denotes the spot color.
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.

Property name	Data Type	Description
GroupParent	Group	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>Constants.FV_CAP_BUTT (0x00)</li> <li>Constants.FV_CAP_ROUND (0x01)</li> <li>Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FMOject	Denotes the next graphic object in the document.
NextGraphicInFrame	FMOject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FMOject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FMOject	Denotes the next selected graphic object in the document.
ObjectAttributes	Strings	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.



Property name	Data Type	Description
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_TR_NONE (0x01)</li> <li>• Constants.FV_TR_CONTOUR (0x02)</li> <li>• Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.
TintPercent	int	Specifies the tint percentage.
Unique	int	Indicates the graphic object's internal unique number.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## Polyline methods

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

## PropIdent

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

## PropIdent methods

[PropIdent](#).

## PropVal

Property name	Data Type	Description
propIdent	<a href="#">PropIdent</a>	The property identifier.
propVal	<a href="#">TypedVal</a>	The property value.

## PropVal methods

[PropVal](#).

## PropVals

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

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

## PropVals methods

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

## Rectangle

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

Property name	Data Type	Description
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_ARROW_STICK (0x1)</li> <li>• Constants.FV_ARROW_HOLLOW (0x2)</li> <li>• Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	Color	Denotes the spot color.
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	Group	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>

Property name	Data Type	Description
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is –216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is –216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FObject	Denotes the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Denotes the next selected graphic object in the document.
ObjectAttributes	Strings	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are:  <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows:  <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.

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

## Rectangle methods

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

## RefPage

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

## RefPage methods

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

## RoundRect

Property name	Data Type	Description
Radius	int	Radius of corner; 0 for a square corner.
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>Constants.FV_ARROW_STICK (0x1)</li> <li>Constants.FV_ARROW_HOLLOW (0x2)</li> <li>Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	Color	Denotes the spot color.
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>Constants.FV_FILL_BLACK (0)</li> <li>Constants.FV_FILL_WHITE (7)</li> <li>Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	Group	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.

Property name	Data Type	Description
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FMOBJECT	Denotes the next graphic object in the document.
NextGraphicInFrame	FMOBJECT	Denotes the next graphic object in the frame.
NextGraphicInGroup	FMOBJECT	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FMOBJECT	Denotes the next selected graphic object in the document.
ObjectAttributes	Strings	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>

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

## RoundRect methods

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

## Row

Property name	Data Type	Description
CondFmtIsShown	int	1 if the condition is shown.
Element	<a href="#">Element</a>	The ID of the element containing the row in a FrameMaker document.
FirstCellInRow	<a href="#">Cell</a>	First cell in row (FO_Cell ID).
Height	int	Height of the row.
InCond	Objects	Condition tags for row (array of FO_CondFmt IDs).
LocX	int	Offset from the left side of the text frame containing the row.



Property name	Data Type	Description
LocY	int	Offset from the top of the page frame containing the row.
NextRowInTbl	<a href="#">Row</a>	Next row (FO_Row ID) in the table.
PrevRowInTbl	<a href="#">Row</a>	Previous row (FO_Row ID) in the table.
RowsShown	int	1 if the conditional row is shown.
RowKeepWithNext	int	1 if Keep With Next Row is enabled.
RowKeepWithPrev	int	1 if Keep With Previous Row is enabled.
RowMaxHeight	int	Maximum row height.
RowMinHeight	int	Minimum row height.
RowStart	int	Denotes the row placement. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_ROW_ANYWHERE (0)</li> <li>• Constants.FV_ROW_TOP_OF_COL (1)</li> <li>• Constants.FV_ROW_TOP_OF_PAGE (2)</li> <li>• Constants.FV_ROW_TOP_OF_LEFT_PAGE (3)</li> <li>• Constants.FV_ROW_TOP_OF_RIGHT_PAGE (4)</li> </ul>
RowTbl	<a href="#">Tbl</a>	Table containing the row (FO_Tbl ID).
RowType	int	Type of row. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_ROW_HEADING (0)</li> <li>• Constants.FV_ROW_BODY (1)</li> <li>• Constants.FV_ROW_FOOTING (2)</li> </ul>
SepOverride	<a href="#">Color</a>	Color separation format override (FO_Color ID).
StyleOverrides	int	Style condition indicators for conditional text: <ul style="list-style-type: none"> <li>• Constants.FV_CS_NO_OVERRIDE (0x00)</li> <li>• Constants.FV_CS_OVERLINE (0x01)</li> <li>• Constants.FV_CS_STRIKETHROUGH (0x02)</li> <li>• Constants.FV_CS_SINGLE_UNDERLINE (0x04)</li> <li>• Constants.FV_CS_DOUBLE_UNDERLINE (0x08)</li> </ul> <p>All style condition indicators are represented as hatched lines for the table rows.</p>
UseSepOverride	int	1 if the <code>SepOverride</code> property overrides default from the table.
Width	int	Width of the row.

## Row methods

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

## Rubi

Property name	Data Type	Description
Element	<a href="#">Element</a>	The object handle of the associated FO_Element for the rubi group element, if the rubi group is in a structured document.
OyamojiTextRange	<a href="#">TextRange</a>	The text range that the oyamoji text encompasses.
NextRubiInDoc	<a href="#">Rubi</a>	The next instance of a rubi composite (FO_Rubi ID) in the document..
RubiElement	<a href="#">Element</a>	The object handle of the associated FO_Element for the rubi element, if the rubi group is in a structured document.
RubiTextRange	<a href="#">TextRange</a>	The text range that the rubi text encompasses.
Unique	int	The rubi composite's unique ID.

### Rubi methods

[Delete](#), [GetProps](#), [GetText2](#), [ObjectValid](#), [SetProps](#).

## RulingFmt

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

### RulingFmt methods

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

## Strings

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

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

## Strings methods

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

## SubCol

Property name	Data Type	Description
ContentHeight	int	The distance between the top of the column and the baseline of the last line in the column.
FirstAFrame	<a href="#">AFrame</a>	First anchored frame in the column (FO_AFrame ID).
FirstCell	<a href="#">Cell</a>	First table cell in the column (FO_Cell ID).
FirstFn	<a href="#">Fn</a>	First footnote in the column (FO_Fn ID).
FirstPgf	<a href="#">Pgf</a>	First paragraph in the column (FO_Pgf ID).
FrameParent	<a href="#">TextFrame</a>	ID of text frame that contains the column (FO_TextFrame ID).
Height	int	Column Height.
LastAFrame	<a href="#">AFrame</a>	Last anchored frame in the column (FO_AFrame ID).
LastCell	<a href="#">Cell</a>	Last table cell in the column (FO_Cell ID).
LastFn	<a href="#">Fn</a>	Last footnote in the column (FO_Fn ID).
LastPgf	<a href="#">Pgf</a>	Last paragraph in the column (FO_Pgf ID).
LocX	int	Offset from left side of the text frame that contains the column.
LocY	int	Offset from top of text frame that contains the column.
NextSubCol	<a href="#">SubCol</a>	Next column in the flow.
Overflowed	int	1 if the text frame containing the column has Autoconnect turned off and text overflows the column.
ParentTextFrame	<a href="#">TextFrame</a>	ID of text frame that contains the column (FO_TextFrame ID).
PrevSubCol	<a href="#">SubCol</a>	Previous column in the flow.
Unique	int	Text column's unique ID.
Width	int	Column width.

## SubCol methods

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

## Tab

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

## Tab methods

[Tab](#).

## Tabs

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

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

## Tabs methods

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

# Tbl

Property name	Data Type	Description
ContentHeight	int	The height of the table title.
Locked	int	1 if the table is part of a text inset that retains formatting information from the source document. The table is not affected by global formatting performed on the document.
OrphanRows	int	Number of orphan rows.
Overflowed	int	1 if the table has cells that are not shown because they extend beyond the text frame boundaries.
TblAlignment	int	Horizontal placement of table. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ALIGN_TBL_LEFT (0)</li> <li>Constants.FV_ALIGN_TBL_CENTER (1)</li> <li>Constants.FV_ALIGN_TBL_RIGHT (2)</li> </ul>
TblCellBottomMargin	int	Default bottom cell margin for the table.
TblCellLeftMargin	int	Default left cell margin for the table.
TblCellRightMargin	int	Default right cell margin for the table.
TblCellTopMargin	int	Default top cell margin for the table.
TblLeftIndent	int	Left indent for the table.
TblInLockedTi	bool	1 if the table is in a locket text inset. You should not use the ESTK to delete table elements when the table is in a locket text inset.
TblPlacement	int	Vertical placement of table on page. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TBL_ANYWHERE (0)</li> <li>Constants.FV_TBL_TOP_OF_COL (1)</li> <li>Constants.FV_TBL_TOP_OF_PAGE (2)</li> <li>Constants.FV_TBL_TOP_OF_LEFT_PAGE (3)</li> <li>Constants.FV_TBL_TOP_OF_RIGHT_PAGE (4)</li> <li>Constants.FV_TBL_FLOAT (5)</li> </ul>
TblRightIndent	int	Right indent for the table.
TblSpaceAbove	int	Vertical space above the table.
TblSpaceBelow	int	Vertical space below the table.
TextLoc	<a href="#">TextLoc</a>	The text location of the table's anchor.
Unique	int	The table's unique ID.
FirstRowInTbl	<a href="#">Row</a>	First row in the table (FO_Row ID).
LastRowInTbl	<a href="#">Row</a>	Last row in the table (FO_Row ID).
NextTblInDoc	<a href="#">Tbl</a>	Next table (FO_Tbl ID) in the document.

Property name	Data Type	Description
TblCatalogEntry	int	1 if the table's format is in the Table Catalog.
TblColWidths	<a href="#">Metrics</a>	List of column widths.
TblNumbering	int	Direction of autonumbering for the table. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_TBL_NUM_BY_ROW (0)</li> <li>• Constants.FV_TBL_NUM_BY_COL (1)</li> </ul>
TblNumCols	int	Number of columns in the table.
TblNumRows	int	Number of rows in the table.
TblTag	string	Name of the table format.
TblWidth	int	Horizontal width of the table.
TblBodyRowRuling	<a href="#">RulingFmt</a>	Ruling applied to body rows specified by TblBodyRowRulingPeriod (FO_RulingFmt ID).
TblBodyRowRulingPeriod	int	The periodicity of the ruling specified by TblBodyRowRuling. For example, if TblBodyRowRulingPeriod is set to 3, the ruling specified by TblBodyRowRuling is applied to every third row.
TblBottomRuling	int	Ruling applied to the bottom of the table (FO_RulingFmt ID).
TblColRuling	<a href="#">RulingFmt</a>	Ruling applied to table columns specified by TblColRulingPeriod (FO_RulingFmt ID).
TblColRulingPeriod	int	The periodicity of the ruling specified by TblColRuling. For example, if TblColRulingPeriod is set to 2, the ruling specified by TblColRuling is applied to every other column.
TblHFRowRuling	<a href="#">RulingFmt</a>	Ruling for table heading and footing rows (FO_RulingFmt ID).
TblHFSeparatorRuling	<a href="#">RulingFmt</a>	Separator ruling for table heading and footing rows (FO_RulingFmt ID).
TblLastBodyRuling	int	1 if Draw Bottom Ruling on Last Sheet Only is enabled (FO_RulingFmt ID).
TblLeftRuling	<a href="#">RulingFmt</a>	Ruling for the left side of the table (FO_RulingFmt ID).
TblOtherBodyRowRuling	<a href="#">RulingFmt</a>	Ruling for body rows that are not specified by TblBodyRowRulingPeriod (FO_RulingFmt ID).
TblOtherColRuling	<a href="#">RulingFmt</a>	Ruling for columns that are not specified by TblColRulingPeriod (FO_RulingFmt ID).
TblRightRuling	<a href="#">RulingFmt</a>	Ruling for the right side of the table (FO_RulingFmt ID).
TblTopRuling	<a href="#">RulingFmt</a>	Ruling for the top of the table (FO_RulingFmt ID).
BottomRowSelection	<a href="#">Row</a>	Bottom body row in selection, if table is selected (FO_Row ID).

Property name	Data Type	Description
LeftColNum	int	Number of the leftmost selected column, if a table is selected (columns are numbered from left to right, starting with 0).
RightColNum	int	Number of rightmost selected column, if a table is selected (columns are numbered from left to right, starting with 0).
TblTitleSelected	int	1 if table title is selected.
TopRowSelection	Row	Top row in selection, if table is selected (FO_Row ID).
TblBodyFirstColor	Color	First spot color for table body (FO_Color ID).
TblBodyFirstFill	int	First fill pattern for table body.
TblBodyFirstPeriod	int	Number of columns or body rows to which the first fill pattern (specified by TblBodyFirstFill) is applied.
TblBodyNextColor	Color	Exception color for columns or body rows (FO_Color ID).
TblBodyNextFill	int	Exception fill pattern for table body.
TblBodyNextPeriod	int	Number of columns or body rows to which the exception fill pattern (specified by TblBodyNextFill) is applied.
TblBodyShadeBy	int	1 if Shade By is set to Columns; 0 if Shade By is set to Rows.
TblHFColor	Color	Color for table heading and footing.
TblHFFill	int	Fill pattern for table heading and footing (integer percentage).
Element	Element	Denotes the element associated with the table.
TblBodyElement	Element	Specifies the element containing the table's body rows.
TblElement	Element	Specifies the element containing the table.
TblFooterElement	Element	Denotes the element containing the table's footer rows.
TblHeaderElement	Element	Denotes the element containing the table's header rows.
TblTitleElement	Element	Indicates the element containing the table title.
FirstPgf	Pgf	The first paragraph in the title, if the table has a title.
HighestLevelElement	Element	Specifies the title's highest-level element if the table is in a structured document and has a title.  <i>Note: HighestLevelElement is obsolete but is supported for backward compatibility.</i>

Property name	Data Type	Description
LastPgf	Pgf	The last paragraph in the title, if the table has a title.
TblTitleGap	int	Gap between the title and top or bottom row of the table.
TblTitlePosition	int	The placement of the table title. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TBL_NO_TITLE (0) - Table has no title</li> <li>Constants.FV_TBL_TITLE_ABOVE (1) - The title appears above the table</li> <li>Constants.FV_TBL_TITLE_BELOW (2) - The title appears below the table</li> </ul>

## Tbl methods

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

## TblFmt

Property name	Data Type	Description
OrphanRows	int	Number of orphan rows.
TblAlignment	int	Horizontal placement of table. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_ALIGN_TBL_LEFT (0)</li> <li>Constants.FV_ALIGN_TBL_CENTER (1)</li> <li>Constants.FV_ALIGN_TBL_RIGHT (2)</li> </ul>
TblCellBottomMargin	int	Bottom margin of table cell.
TblCellLeftMargin	int	Left margin of the table cell.
TblCellRightMargin	int	Right margin of the table cell.
TblCellTopMargin	int	Top margin of the table cell.
TblLeftIndent	int	Left indent of the table cell.
TblPlacement	int	Vertical placement of table on page. The value is one of: <ul style="list-style-type: none"> <li>Constants.FV_TBL_ANYWHERE (0)</li> <li>Constants.FV_TBL_TOP_OF_COL (1)</li> <li>Constants.FV_TBL_TOP_OF_PAGE (2)</li> <li>Constants.FV_TBL_TOP_OF_LEFT_PAGE (3)</li> <li>Constants.FV_TBL_TOP_OF_RIGHT_PAGE (4)</li> <li>Constants.FV_TBL_FLOAT (5)</li> </ul>
TblRightIndent	int	Right indent of the table.
TblSpaceAbove	int	Vertical space above the table.
TblSpaceBelow	int	Vertical space below the table.



Property name	Data Type	Description
TblTitleGap	int	Gap between title and top or bottom row.
TblTitlePosition	int	The placement of the table title. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_TBL_NO_TITLE (0) - Table has no title</li> <li>• Constants.FV_TBL_TITLE_ABOVE (1) - The title appears above the table</li> <li>• Constants.FV_TBL_TITLE_BELOW (2) - The title appears below the table</li> </ul>
TblCatalogEntry	int	1 if format is in the Table Catalog.
Name	string	Name of the paragraph format of the table title.
NextTblFmtInDoc	<a href="#">TblFmt</a>	Next table format in the document (FO_TblFmt ID).
TblNumbering	int	Direction of autonumbering for the table. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_TBL_NUM_BY_ROW (0)</li> <li>• Constants.FV_TBL_NUM_BY_COL (1)</li> </ul>
TblTag	string	Name of the table format
TblInitNumBodyRows	int	Number of body rows for new table
TblInitNumCols	int	Number of columns for new table
TblInitNumFRows	int	Number of footing rows for new table
TblInitNumHRows	int	Number of heading rows for new table
TblBodyRowRuling	<a href="#">RulingFmt</a>	Ruling for body rows that are not specified by TblBodyRowRulingPeriod (FO_RulingFmt ID).
TblBodyRowRulingPeriod	int	The periodicity of the ruling specified by TblOtherBodyRowRuling. For example, if TblBodyRowRulingPeriod is set to 3, the ruling specified by TblOtherBodyRowRuling is applied to every third row.
TblBottomRuling	<a href="#">RulingFmt</a>	Ruling for the bottom of the table (FO_RulingFmt ID).
TblColRuling	<a href="#">RulingFmt</a>	Ruling for columns that are not specified by TblColRulingPeriod (FO_RulingFmt ID).
TblColRulingPeriod	int	The periodicity of the ruling specified by TblOtherColRuling. For example, if TblColRulingPeriod is set to 2, the ruling specified by TblOtherColRuling is applied to every other column.
TblHFRowRuling	<a href="#">RulingFmt</a>	Ruling for the heading and footing rows (FO_RulingFmt ID).
TblHFSeparatorRuling	<a href="#">RulingFmt</a>	Separator ruling for the table heading and footing rows (FO_RulingFmt ID).
TblLastBodyRuling	int	1 if Draw Bottom Ruling on Last Sheet Only is enabled (FO_RulingFmt ID).
TblLeftRuling	<a href="#">RulingFmt</a>	Ruling for the left side of the table (FO_RulingFmt ID).

Property name	Data Type	Description
TblOtherBodyRowRuling	<a href="#">RulingFmt</a>	Ruling applied to body rows specified by TblBodyRowRulingPeriod (FO_RulingFmt ID).
TblOtherColRuling	<a href="#">RulingFmt</a>	Ruling applied to table columns specified by TblColRulingPeriod (FO_RulingFmt ID).
TblRightRuling	<a href="#">RulingFmt</a>	Ruling for the right side of the table (FO_RulingFmt ID).
TblTopRuling	<a href="#">RulingFmt</a>	Ruling for the top of the table (FO_RulingFmt ID).
TblBodyFirstColor	<a href="#">Color</a>	First spot color for table body (FO_Color ID)
TblBodyFirstFill	int	First fill pattern for table body
TblBodyFirstPeriod	int	Number of columns or body rows to which the first fill pattern (specified by TblBodyFirstFill) is applied
TblBodyNextColor	<a href="#">Color</a>	Exception color for columns or body rows (FO_Color ID)
TblBodyNextFill	int	Exception fill pattern for table body
TblBodyNextPeriod	int	Number of columns or body rows to which the exception fill pattern (specified by TblBodyNextFill) is applied
TblBodyShadeBy	int	1 if Shade By is set to Columns; 0 if Shade By is set to Rows
TblHFColor	<a href="#">Color</a>	Color for table heading and footing
TblHFFill	int	Fill pattern for table heading and footing (integer percentage)

## TblFmt methods

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

## TextFrame

Property name	Data Type	Description
ColGapWidth	int	Gap between columns (0 to 50 inches).
ColumnsAreBalanced	int	1 if terminal and underfilled columns in the flow are balanced.
FirstAFrame	<a href="#">AFrame</a>	First anchored frame in the text frame (FO_AFrame ID).
FirstCell	<a href="#">Cell</a>	First table cell in the text frame (FO_Cell ID).
FirstFn	<a href="#">Fn</a>	First footnote in the text frame (FO_Fn ID).
FirstPgf	<a href="#">Pgf</a>	First paragraph in the text frame (FO_Pgf ID).
FirstSubCol	<a href="#">SubCol</a>	First column in the text frame (FO_SubCol ID).
Flow	<a href="#">Flow</a>	Flow containing the text frame (FO_Flow ID).
GraphicsButton	int	1 if the text frame is a hypertext button.
LastAFrame	<a href="#">AFrame</a>	Last anchored frame in the text frame (FO_AFrame ID).

Property name	Data Type	Description
LastCell	Cell	Last table cell in the text frame (FO_Cell ID).
LastFn	Fn	Last footnote in the text frame (FO_Fn ID).
LastPgf	Pgf	Last paragraph in the text frame (FO_Pgf ID).
LastSubCol	SubCol	Last column in the text frame (FO_SubCol ID).
NextTextFrameInFlow	TextFrame	Next text frame in the flow (FO_TextFrame ID).
NumColumns	int	The number of columns in the underlying column grid (1–10).
PrevTextFrameInFlow	TextFrame	Previous text frame in the flow (FO_TextFrame ID).
SideHeadGap	int	Gap between side head area and body text area (0 to 50 inches).
SideHeadPlacement	int	Placement of side heads relative to columns in the text frame: <ul style="list-style-type: none"> <li>• Constants.FV_SH_LEFT (0x00)</li> <li>• Constants.FV_SH_RIGHT (0x01)</li> <li>• Constants.FV_SH_INSIDE (0x02)</li> <li>• Constants.FV_SH_OUTSIDE (0x03)</li> </ul>
SideHeadWidth	int	Width of side head area for the text frame (0 to 50 inches).
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is False.
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_ARROW_STICK (0x1)</li> <li>• Constants.FV_ARROW_HOLLOW (0x2)</li> <li>• Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	Color	Denotes the spot color.

Property name	Data Type	Description
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	Group	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.

Property name	Data Type	Description
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is -216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FObject	Denotes the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Denotes the next selected graphic object in the document.
ObjectAttributes	Strings	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
PrevGraphicInFrame	FObject	Denotes the previous graphic object in the frame.
PrevGraphicInGroup	FObject	Denotes the previous graphic object in the group.
Runaround	int	Specifies whether text can flow around the object and, if so, whether the text follows the contour of the object or a box shape surrounding the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_TR_NONE (0x01)</li> <li>• Constants.FV_TR_CONTOUR (0x02)</li> <li>• Constants.FV_TR_BBOX (0x03)</li> </ul>
RunaroundGap	int	Denotes the width of the runaround gap, if the object is a runaround object.
TailArrow	int	Set to 1 if the arrowhead is enabled at the beginning of the line; Set to 0 otherwise.

Property name	Data Type	Description
TintPercent	int	Specifies the tint percentage.
Unique	int	Indicates the graphic object's internal unique number.
Width	int	Denotes the width of the object. The permissible range is from 0.125 points to 3600 points.

## TextFrame methods

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

## TextItem

Property name	Data Type	Description
offset	int	Offset from the beginning of the paragraph or the text line.
dataType	int	Type of the text item.
sdata	string	Denotes the text item if the text item is a string.
idata	number	ID of the object if the text item is an object.
obj	any	Denotes the text item if the text item is an object.

The type of Text Item is one of the following:

Text Item Type	Text Item Represents	Text Item Data
FTI_TextObjId	The object to which the offsets of all the text items are relative.	FO_Pgf, FO_Cell, FO_TextLine, FO_TiApiClient, FO_TiFlow, FO_TiText, or FO_TiTextTable
FTI_String	A string of characters with the same condition and character format.	A character string
FTI_LineBegin	The beginning of a line.	-
FTI_LineEnd	The end of a line and the line end type.	If the line end is a normal line end, the value is 0; if it is a forced line end, the FTI_HardLineEnd flag is set; if it is a hyphen line end, the FTI_HyphenLineEnd flag is set
FTI_PgfBegin	The beginning of a paragraph.	FO_Pgf
FTI_PgfEnd	The end of a paragraph.	FO_Pgf
FTI_FlowBegin	The beginning of a flow.	FO_Flow
FTI_FlowEnd	The end of a flow.	FO_Flow
FTI_PageBegin	The beginning of a page.	FO_Page
FTI_PageEnd	The end of a page.	FO_Page
FTI_SubColBegin	The beginning of a column.	FO_SubCol
FTI_SubColEnd	The end of a column.	FO_SubCol

<b>Text Item Type</b>	<b>Text Item Represents</b>	<b>Text Item Data</b>
FTI_FrameAnchor	An anchored frame.	FO_AFrame
FTI_FnAnchor	A footnote.	FO_Fn
FTI_TblAnchor	A table.	FO_Tbl
FTI_MarkerAnchor	A marker.	FO_Marker
FTI_XRefBegin	The beginning of a cross-reference.	FO_XRef
FTI_XRefEnd	The end of a cross-reference.	FO_XRef
FTI_TextFrameBegin	The beginning of a text frame.	FO_TextFrame
FTI_TextFrameEnd	The end of a text frame.	FO_TextFrame
FTI_VarBegin	The beginning of a variable.	FO_Var
FTI_VarEnd	The end of a variable.	FO_Var
FTI_ElementBegin	The beginning of a container structural element.	FO_Element
FTI_ElementEnd	The end of a container structural element.	FO_Element
FTI_ElemPrefixBegin	The beginning of an element's prefix.	FO_Element
FTI_ElemPrefixEnd	The end of an element's prefix.	FO_Element
FTI_ElemSuffixBegin	The beginning of an element's suffix.	FO_Element
FTI_ElemSuffixEnd	The end of an element's suffix.	FO_Element
FTI_CharPropsChange	A change in the text properties.	Flags indicating which properties have changed (see the following table).
FTI_RubiCompositeBegin	The beginning of a rubi composite (and the beginning of oyamoji text).	FO_Rubi
FTI_RubiCompositeEnd	The end of a rubi composite.	FO_Rubi
FTI_RubiTextBegin	The beginning of rubi text (and the end of oyamoji text).	FO_Rubi
FTI_RubiTextEnd	The end of rubi text.	FO_Rubi

The following table lists the flags for `FTI_CharPropsChange`. These flags indicate which properties have changed.

<b>Flags</b>	<b>Meaning</b>
FTF_ALL	OR of all the flags
FTF_ANGLE	The font angle has changed.
FTF_CAPITALIZATION	The capitalization has changed.
FTF_CHANGEBAR	The change bars have changed.
FTF_CHARTAG	The Character Catalog format has changed.
FTF_COLOR	The color has changed.
FTF_CONDITIONTAG	The condition tag has changed.
FTF_ENCODING	The text encoding has changed,
FTF_FAMILY	The font family has changed.

Flags	Meaning
FTF_IIF	An internal flag having to do with Asian text. input. If there is a non-zero value for this flag, a front-end processor is controlling that text and you should not modify the associated text item.
FTF_KERNX	The kern-x characteristic has changed.
FTF_KERNY	The kern-x characteristic has changed.
FTF_LANGUAGE	Character language has changed.
FTF_OUTLINE	The outline characteristic has changed.
FTF_OVERLINE	The overline characteristic has changed.
FTF_PAIRKERN	The pair kerning has changed.
FTF_POSITION	The character position has changed.
FTF_SHADOW	The shadow characteristic has changed.
FTF_SIZE	The font size has changed.
FTF_SPREAD	The font spread has changed.
FTF_STRETCH	The font stretch value has changed.
FTF_STRIKETHROUGH	The strikethrough characteristic has changed.
FTF_TSUME	The Tsume setting has changed.
FTF_UNDERLINING	The underlining has changed.
FTF_VARIATION	The font variation has changed.
FTF_WEIGHT	The font weight has changed.

## TextItems

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

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

### TextItems methods

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



## TextLine

Property name	Data Type	Description
BasePointX	int	Horizontal placement of text line base point relative to left side of the frame.
BasePointY	int	Vertical placement of text line base point relative to top of the frame.
Language	int	<p>Hyphenation and Spell Checking language to use.</p> <ul style="list-style-type: none"> <li>• Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li>• Constants.FV_LANG_ENGLISH (0x01)</li> <li>• Constants.FV_LANG_BRITISH (0x02)</li> <li>• Constants.FV_LANG_GERMAN (0x03)</li> <li>• Constants.FV_LANG_SWISS_GERMAN (0x04)</li> <li>• Constants.FV_LANG_FRENCH (0x05)</li> <li>• Constants.FV_LANG_CANADIAN_FRENCH (0x06)</li> <li>• Constants.FV_LANG_SPANISH (0x07)</li> <li>• Constants.FV_LANG_CATALAN (0x08)</li> <li>• Constants.FV_LANG_ITALIAN (0x09)</li> <li>• Constants.FV_LANG_FINNISH (0x10)</li> <li>• Constants.FV_LANG_SWEDISH (0x11)</li> <li>• Constants.FV_LANG_JAPANESE (0x12)</li> <li>• Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>• Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>• Constants.FV_LANG_KOREAN (0x15)</li> <li>• Constants.FV_LANG_PORTUGUESE (0x0A)</li> <li>• Constants.FV_LANG_BRAZILIAN (0x0B)</li> <li>• Constants.FV_LANG_DANISH (0x0C)</li> <li>• Constants.FV_LANG_DUTCH (0x0D)</li> <li>• Constants.FV_LANG_NORWEGIAN (0x0E)</li> <li>• Constants.FV_LANG_NYNORSK (0x0F)</li> </ul>
TextLineType	int	<p>Type of text line. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_TEXTLINE_LEFT (0)</li> <li>• Constants.FV_TEXTLINE_RIGHT (1)</li> <li>• Constants.FV_TEXTLINE_CENTER (2)</li> <li>• Constants.FV_TEXTLINE_MATH (3)</li> </ul>
Angle	int	Indicates the angle of rotation of the graphic object.

Property name	Data Type	Description
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_ARROW_STICK (0x1)</li> <li>• Constants.FV_ARROW_HOLLOW (0x2)</li> <li>• Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	Color	Denotes the spot color.
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	Group	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.

Property name	Data Type	Description
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is – 216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is –216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FObject	Denotes the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Denotes the next selected graphic object in the document.
ObjectAttributes	Strings	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7).  Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>

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

## TextLine methods

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

## TextLoc

Property name	Data Type	Description
obj	any	The object containing text.
offset	int	Characters from the start of the object.

## TextLoc methods

[TextLoc](#).

## TextRange

Property name	Data Type	Description
beg	<a href="#">TextLoc</a>	The beginning of the text range.
end	<a href="#">TextLoc</a>	The end of the text range.

## TextRange methods

[TextRange](#).

## TiApiClient

Property name	Data Type	Description
TiClientData	string	Data used by the client (for example, an SQL query).
TiClientName	string	The registered name of the client that created the inset.
TiClientSource	string	The name that appears as the source in the Text Inset Properties dialog box.
TiClientType	string	The name that appears as the source type in the Text Inset Properties dialog box.
TiIsUnresolved	int	1 if the inset is unresolved. A client should set this property to 0 if it is unable to resolve the inset.
ImportHint	string	Record identifying the filter used to import the text. The FrameMaker product uses this record to find the filter to use when updating the inset.
TiLocked	int	1 if the inset is locked. To change an inset's contents, you must unlock it. Always relock an inset after you have finished changing its contents.
Name	string	A name assigned to the inset by an FDK client. It is not automatically assigned by the FrameMaker product.
NextTiInDoc	FObject	The ID of the next text inset in the list of text insets in the document ( <code>Constants.FO_TiApiClient</code> , <code>Constants.FO_TiText</code> , <code>Constants.FO_TiTextTable</code> , or <code>Constants.FO_TiFlow ID</code> ).
TextRange	<a href="#">TextRange</a>	The text range, in the document containing the text inset, occupied by the text inset.
TiAutomaticUpdate	int	1 if the inset is updated automatically. <code>TiAutomaticUpdate</code> has no effect if the document's <code>DontUpdateTextInsets</code> property is set to <code>True</code> .
TiFile	string	Path of the source file.
TiFileModDate	string	The modification date of the text inset's source file.
LastUpdate	int	Time when the inset was last updated, expressed in seconds since 1 January, 1970.
Unique	int	The text inset's UID.

## TiApiClient methods

[ConvertToText](#), [Delete](#), [DeletePropByName](#), [DeleteTextInsetContents](#), [GetProps](#), [GetText](#), [ObjectValid](#), [SetProps](#), [UpdateTextInset](#).

## TiFlow

Property name	Data Type	Description
TiFlowName	string	The name of the imported flow if TiMainFlow is False.
TiFlowPageSpace	int	The type of pages the imported flow is on: <ul style="list-style-type: none"> <li>Constants.FV_BODY_PAGE(0x00)</li> <li>Constants.FV_REFERENCE_PAGE (0x02)</li> </ul>
TiFormat	int	Source of the imported text's format: <ul style="list-style-type: none"> <li>Constants.FV_SourceDoc (0) - The text is formatted with formats from the source document.</li> <li>Constants.FV_EnclosingDoc (1) - The text is formatted with formats from the document into which it is imported.</li> <li>Constants.FV_PlainText (2) - The text is formatted as plain text.</li> </ul>
TiMainFlow	int	1 if the inset text is imported from the main flow of the source document.
TiRemovePageBreaks	int	1 if page breaks are removed from the text when TiFormat is set to Constants.FV_EnclosingDoc(1).
TiRemoveOverrides	int	1 if format overrides are removed from the text when TiFormat is set to Constants.FV_EnclosingDoc(1).
ImportHint	string	Record identifying the filter used to import the text. The FrameMaker product uses this record to find the filter to use when updating the inset.
TiLocked	int	1 if the inset is locked. To change an inset's contents, you must unlock it. Always relock an inset after you have finished changing its contents.
Name	string	A name assigned to the inset by an FDK client. It is not automatically assigned by the FrameMaker product.
NextTiInDoc	FObject	The ID of the next text inset in the list of text insets in the document (Constants.FO_TiApiClient (64), Constants.FO_TiText (62), Constants.FO_TiTextTable (63), or Constants.FO_TiFlow (61) ID).
TextRange	<a href="#">TextRange</a>	The text range, in the document containing the text inset, occupied by the text inset.
TiAutomaticUpdate	int	1 if the inset is updated automatically.  TiAutomaticUpdate has no effect if the document's DontUpdateTextInsets property is set to True.
TiFile	string	Path of the source file.

Property name	Data Type	Description
TiFileModDate	string	The modification date of the text inset's source file.
LastUpdate	int	Time when the inset was last updated, expressed in seconds since 1 January, 1970.
Unique	int	The text inset's UID.

## TiFlow methods

[ConvertToText](#), [Delete](#), [DeletePropByName](#), [DeleteTextInsetContents](#), [GetProps](#), [GetText](#), [ObjectValid](#), [SetProps](#), [UpdateTextInset](#).

## TiText

Property name	Data Type	Description
TiEOLisEOP	int	1 if line ends in the imported text file are treated as paragraph ends.
TiTextEncoding	string	The <code>ImportHintString</code> for the text inset. If this property is not set to <code>Constants.FO_TiText (62)</code> or <code>Constants.FO_TiTextTable (63)</code> , the string is null.
ImportHint	string	Record identifying the filter used to import the text. The FrameMaker product uses this record to find the filter to use when updating the inset.
TiLocked	int	1 if the inset is locked. To change an inset's contents, you must unlock it. Always relock an inset after you have finished changing its contents.
Name	string	A name assigned to the inset by an ESTK client. It is not automatically assigned by the FrameMaker product.
NextTiInDoc	FMOBJECT	The ID of the next text inset in the list of text insets in the document ( <code>Constants.FO_TiApiClient (64)</code> , <code>Constants.FO_TiText (62)</code> , <code>Constants.FO_TiTextTable (63)</code> , or <code>Constants.FO_TiFlow (61)</code> ID).
TextRange	<a href="#">TextRange</a>	The text range, in the document containing the text inset, occupied by the text inset.
TiAutomaticUpdate	int	1 if the inset is updated automatically.  <code>TiAutomaticUpdate</code> has no effect if the document's <code>DontUpdateTextInsets</code> property is set to <code>True</code> .
TiFile	string	Path of the source file.
TiFileModDate	string	The modification date of the text inset's source file.
LastUpdate	int	Time when the inset was last updated, expressed in seconds since 1 January, 1970.
Unique	int	The text inset's UID.

## TiText methods

[ConvertToText](#), [Delete](#), [DeletePropByName](#), [DeleteTextInsetContents](#), [GetProps](#), [GetText](#), [ObjectValid](#), [SetProps](#), [UpdateTextInset](#).

## TiTextTable

Property name	Data Type	Description
TiByRows	int	1 if each paragraph in the imported text is converted to a row of table cells; False if each paragraph in the imported text is converted to a table cell.
TiTblTag	string	The table format tag of the imported table
TiHeadersEmpty	int	1 if the imported text is not used to fill the heading rows.
TiNumSeparators	int	If <code>TiSeparator</code> specifies a space, the number of spaces used as a separator to parse the text into table cells.
TiSeparator	string	If <code>TiByRows</code> is True, a string specifying a separator, such as a tab, used to parse the text into table cells.
TiNumCols	int	If <code>TiByRows</code> is False, the number of columns in the table.
TiNumHeaderRows	int	The number of heading rows in the table.
TiTextEncoding	string	The <code>ImportHintString</code> for the text inset. If this property is not set to <code>Constants.FO_TiText (62)</code> or <code>Constants.FO_TiTextTable (63)</code> , the string is null.
ImportHint	string	Record identifying the filter used to import the text. The FrameMaker product uses this record to find the filter to use when updating the inset.
TiLocked	int	1 if the inset is locked. To change an inset's contents, you must unlock it. Always relock an inset after you have finished changing its contents.
Name	string	A name assigned to the inset by an FDK client. It is not automatically assigned by the FrameMaker product.
NextTiInDoc	FMOBJECT	The ID of the next text inset in the list of text insets in the document ( <code>Constants.FO_TiApiClient (64)</code> , <code>Constants.FO_TiText (62)</code> , <code>Constants.FO_TiTextTable (63)</code> , or <code>Constants.FO_TiFlow (61) ID</code> ).
TextRange	<a href="#">TextRange</a>	The text range, in the document containing the text inset, occupied by the text inset.
TiAutomaticUpdate	int	1 if the inset is updated automatically.  <code>TiAutomaticUpdate</code> has no effect if the document's <code>DontUpdateTextInsets</code> property is set to True.
TiFile	string	Path of the source file.



Property name	Data Type	Description
TiFileModDate	string	The modification date of the text inset's source file.
LastUpdate	int	Time when the inset was last updated, expressed in seconds since 1 January, 1970.
Unique	int	The text inset's UID.

## TiTextTable methods

[ConvertToText](#), [Delete](#), [DeletePropByName](#), [DeleteTextInsetContents](#), [GetProps](#), [GetText](#), [ObjectValid](#), [SetProps](#), [UpdateTextInset](#).

## TypedVal

Property name	Data Type	Description
valType	int	The type of value. See the following table.
sval	string	String value.
ssval	<a href="#">Strings</a>	Set of strings.
mssval	<a href="#">Metrics</a>	Set of metrics.
psval	<a href="#">Points</a>	Set of points.
tsval	<a href="#">Tabs</a>	Set of tabs.
tlval	<a href="#">TextLoc</a>	Text location.
trval	<a href="#">TextRange</a>	Text range.
adssval	<a href="#">AttributeDefs</a>	TBD
assval	<a href="#">Attribute</a>	TBD
cssval	<a href="#">ElementCatalogEntries</a>	Element Catalog values.
isval	<a href="#">Ints</a>	Set of integers.
osval	<a href="#">Objects</a>	TBD
uisval	<a href="#">UInts</a>	Set of unsigned integers.
valsval	<a href="#">TypedVals</a>	Set of TypedVal entries.
obj	any	Set of objects.
ival	int	An integer.

## TypedVals

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

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

## TypedVals methods

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

## UBytes

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

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

## UBytes methods

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

## UInts

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

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

## UInts methods

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

## UnanchoredFrame

Property name	Data Type	Description
Name	string	The name of the frame.
PageFramePage	FMOBJECT	The page that the unanchored frame belongs to, if the unanchored frame is a page frame. The page can be one of:FO_HiddenPage, FO_BodyPage, FO_MasterPage, or FO_RefPage ID.
FirstGraphicInFrame	FMOBJECT	First object in the frame (backmost object).
LastGraphicInFrame	FMOBJECT	Last object in the frame (frontmost object).
Angle	int	Indicates the angle of rotation of the graphic object.
ArrowBaseAngle	int	Denotes the arrowhead base angle in degrees.

Property name	Data Type	Description
ArrowLength	int	Denotes the arrowhead length (always rounded down to the nearest 1/256 point).
ArrowScaleFactor	int	Indicates the factor by which the arrowhead is scaled as line width changes (always rounded down to nearest 1/16 point). It is not used if the ArrowScaleHead property is <code>False</code> .
ArrowScaleHead	int	Set to 1 if arrowhead is scaled as the line width changes. Set to 0, otherwise.
ArrowTipAngle	int	Denotes the arrowhead tip angle in degrees.
ArrowType	int	Denotes the style of the arrowhead. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_ARROW_STICK (0x1)</li> <li>• Constants.FV_ARROW_HOLLOW (0x2)</li> <li>• Constants.FV_ARROW_FILLED (0x3)</li> </ul>
BorderWidth	int	Indicates the border width specified in points. The permissible range is 0.015 pt to 360 points.
Color	Color	Denotes the spot color.
Dash	Metrics	Specifies a dash pattern that is repeated for the length of an object's border. The pattern is stored in a MetricsT structure. The 0th element of the MetricsT.MetricsT_val array stores the length of the first dash; the 1st element stores the following space; the 2nd element stores the next dash; and so on for an even number of elements.
Fill	int	Denotes the fill pattern (numbers between 0 and 15). Constants are provided for fill patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>
FrameParent	FObject	Denotes the frame containing the graphic object.
GraphicCantBeSelected	int	Set to 1 if the graphic object cannot be selected. Set to 0, otherwise.
GraphicsSelected	int	Set to 1, if the graphic object is selected. Else, set to 0.
GroupParent	Group	Specifies the group to which the object belongs. Note: Anchored frames do NOT have this property.
HeadArrow	int	Set to 1 if the line has an arrow head. Set to 0, otherwise.
Height	int	Denotes the height of the object denoted in points. The permissible range is from 0.125 points to 3600 points.
HotspotCmdStr	StringT	The command string for a hotspot. This must be a valid hypertext command string.
HotspotTitle	StringT	The tooltip text for the hotspot in the outputs that support it (for example HTML). This property is Optional.

Property name	Data Type	Description
IsHotspot	Bool	Whether the object is a hotspot or not. If this property is turned off, the object is no longer a hotspot even if command string is non-empty.
LineCap	int	Specifies the type of the end of the line. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_CAP_BUTT (0x00)</li> <li>• Constants.FV_CAP_ROUND (0x01)</li> <li>• Constants.FV_CAP_SQUARE (0x02)</li> </ul>
LocX	int	Specifies the distance of the object from the left side of the parent frame (in inches). The permissible range is – 216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the left side of the page frame. You cannot set this value manually for anchored frames.
LocY	int	Distance of the object from the top of the parent frame (in inches) The permissible range is –216 inches to 216 inches.  If the graphic object is an anchored frame, the distance is automatically calculated from the top of the page frame. You cannot set this value manually for anchored frames.
NextGraphicInDoc	FObject	Denotes the next graphic object in the document.
NextGraphicInFrame	FObject	Denotes the next graphic object in the frame.
NextGraphicInGroup	FObject	Denotes the next graphic object in the group.
NextSelectedGraphicInDoc	FObject	Denotes the next selected graphic object in the document.
ObjectAttributes	Strings	A list of strings. Each string expresses an attribute that is specified for an anchored frame in the Object Properties > Object Attributes dialog box. Each string is in the form of <i>string_text</i> . You can use the backslash to escape characters such as \n, \r, or \t for newline, carriage return, and tab, respectively. For a backslash character in the string text, type \\.
Overprint	int	Specifies the overprint settings for the object. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_KNOCKOUT (0x00)</li> <li>• Constants.FV_OVERPRINT (0x01)</li> <li>• Constants.FV_FROMCOLOR (0x02)</li> </ul>
Pen	int	Specifies the pen pattern (numbers between 0 and 7). Constants are provided for pen patterns 0, 7 and 15, as follows: <ul style="list-style-type: none"> <li>• Constants.FV_FILL_BLACK (0)</li> <li>• Constants.FV_FILL_WHITE (7)</li> <li>• Constants.FV_FILL_CLEAR (15)</li> </ul>

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

## UnanchoredFrame methods

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

## Var

Property name	Data Type	Description
Element	<a href="#">Element</a>	The element associated with the variable, if the variable is in a FrameMaker document.
Locked	int	1 if the variable is included in a text inset that gets its formatting from the source document. The variable is not affected by global formatting performed on the document.
NextVarInDoc	<a href="#">Var</a>	Next variable instance (FO_Var ID) in the document.
TextRange	<a href="#">TextRange</a>	The text range the variable instance encompasses.
VarFmt	<a href="#">VarFmt</a>	The variable instance's format (FO_VarFmt ID).
Unique	int	The variable's unique ID.

## Var methods

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

# VarFmt

Property name	Data Type	Description
Fmt	string	The variable format definition; the building blocks and text strings used to create a variable instance with the variable format.
Name	string	The variable format's name.
NextVarFmtInDoc	<a href="#">VarFmt</a>	Next variable format (FO_VarFmt ID) in the document's list of variable formats.
SystemVar	int	<p>The variable format's type.</p> <p>The following types specify system variable formats:</p> <ul style="list-style-type: none"> <li>• Constants.FV_VAR_CURRENT_PAGE_NUM (1)</li> <li>• Constants.FV_VAR_PAGE_COUNT (2)</li> <li>• Constants.FV_VAR_CURRENT_DATE_LONG (3)</li> <li>• Constants.FV_VAR_CURRENT_DATE_SHORT (4)</li> <li>• Constants.FV_VAR_MODIFICATION_DATE_LONG (5)</li> <li>• Constants.FV_VAR_MODIFICATION_DATE_SHORT (6)</li> <li>• Constants.FV_VAR_CREATION_DATE_LONG (7)</li> <li>• Constants.FV_VAR_CREATION_DATE_SHORT (8)</li> <li>• Constants.FV_VAR_FILE_NAME_LONG (9)</li> <li>• Constants.FV_VAR_FILE_NAME_SHORT (10)</li> <li>• Constants.FV_VAR_HEADER_FOOTER_1 (11) through Constants.FV_VAR_HEADER_FOOTER_4 (14)</li> <li>• Constants.FV_VAR_TABLE_CONTINUATION (15)</li> <li>• Constants.FV_VAR_TABLE_SHEET (16)</li> <li>• Constants.FV_VAR_HEADER_FOOTER_5 (19) through Constants.FV_VAR_HEADER_FOOTER_12 (26)</li> </ul> <p>The following type specifies user defined formats:</p> <p>Constants.FV_VAR_USER_VARIABLE (0) - A user defined variable format.</p>

## VarFmt methods

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

# XRef

Property name	Data Type	Description
Element	<a href="#">Element</a>	The associated element, if the cross reference is in a FrameMaker document.
Locked	int	1 if the cross-reference is part of a text inset that retains formatting information from the source document.  The cross-reference is not affected by global formatting performed on the document.
NextXRefInDoc	<a href="#">XRef</a>	Next cross-reference instance in document (FO_XRef ID).
TextRange	<a href="#">TextRange</a>	Text range that the cross-reference instance encompasses.
Unique	int	The cross-reference's unique ID.
XRefFmt	<a href="#">XRefFmt</a>	ID of the cross-reference's format (FO_XrefFmt ID).
XRefFile	string	The filename of the file containing the cross-reference source. If the cross-reference source is in the same document as the cross reference, the filename is an empty string ("").
XRefsUnresolved	int	1 if FrameMaker was unable to resolve the cross-reference the last time it updated crossreferences.  <i>Note: This property is set only when FrameMaker updates cross-references. Changes to the document, in and of themselves, do not affect this property.</i>
XRefSrcText	string	If <code>XRefSrcIsElem</code> is <code>False</code> , the <code>XRefSrcText</code> property is a string specifying <code>UID:pgf_tag:text</code> , where <code>UID</code> represents the unique ID for the paragraph, <code>pgf_tag</code> is the name of the paragraph format, and <code>text</code> is the text content of the paragraph. If <code>XRefSrcIsElem</code> is <code>True</code> , the <code>XRefSrcText</code> property is a string specifying <code>UID:src_name:text</code> , where <code>UID</code> is the value of the ID attribute of the source element, <code>name</code> is the element tag, and <code>text</code> is text content of the source element.
XRefSrcIsElem	int	1 if the cross-reference source is a FrameMaker structural element.
XRefSrcElemNonUniqueId	string	If <code>XRefSrcIsElem</code> is <code>True</code> , this is a string specifying the value of <code>@id</code> of the source element inside the element identified by the Unique ID in <code>XRefSrcText</code> . This is required in cases where the <code>Id</code> of the source element is not a Unique ID.
XRefAltText	string	Alternate text for display as the cross-reference link text. This is given preference, if present, over the earlier behavior of getting text from the source.
XRefClientName	string	The registered name of the client that created the cross-reference.
XRefClientType	string	The identifier for specifying the source type for the cross-reference.

## XRef methods

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

## XRefFmt

Property name	Data Type	Description
Fmt	string	The cross-reference format (a string that specifies text and building blocks).
Name	string	The cross-reference format's name.
NextXRefFmtInDoc	<a href="#">XRefFmt</a>	Denotes the next cross-reference format (FO_XRefFmt ID).

## XRefFmt methods

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

## app

Property name	Data Type	Description
ActiveBook	<a href="#">Book</a>	The book with input focus (FO_Book ID).
ActiveDoc	<a href="#">Doc</a>	The document with input focus (FO_Doc ID).
ActiveView	string	Sets the current view. The view can be one of: WYSIWYG View Author View XML View
ApplyFormatRules	int	1 if element reformatting is enabled.
AddMarkerTypeToStandardMarkers	string	The name of a marker type to add to the standard list of marker types. Use <code>SetString()</code> to set a marker type name to this property of the <code>Constants.FV_SessionId(0)</code> .
AutoBackup	int	1 if Automatic Backup is enabled.
AutoSave	int	1 if Automatic Save is enabled.
AutoSaveSeconds	int	Time between automatic saves in seconds (60 seconds to 10800 seconds).
Displaying	int	False if screen refresh is completely turned off.
DisplayName	string	The name of the display on which the FrameMaker product session is running.



Property name	Data Type	Description
DefaultKeyCatalog	Doc	Default Key Catalog for the current workflow.  Workflow values: KeyCatalogWorkflowAuthoring KeyCatalogWorkflowPublishing KeyCatalogWorkflowSearching
FirstKeyCatalogInSession	Doc	First Key Catalog in the session.
KeyCatalogWorkflow	int	Current workflow related to Key Catalogs.
IsTempOpenSave	int	Gets whether temporary open/save is in progress. Temporary open/save happens during view switching operations.
ExportFilters	string	List of export filters available in the current session.
FirstCommandInSession	Command	First command in the list of commands in the session (FO_Command ID).
FirstMenuItemInSession	FMObject	First menu item or menu in the list of menus, menu items, and menu item separators in the session (Constants.FO_Command (56), Constants.FO_Menu (55), Constants.FO_MenuItemSeparator (57) ID).
FirstOpenBook	Book	First open book (FO_Book ID) in session.
FirstOpenDoc	Doc	First open document (FO_Doc ID) in session.
BinDir	string	Directory path of \$FMHOME/bin.
CurrentDir	string	Name of the directory from which the FrameMaker product was started
CurrentMenuSet	int	Type of menu set. The value is one of: <ul style="list-style-type: none"><li>• Constants.FV_MENU_QUICK (1)</li><li>• Constants.FV_MENU_COMPLETE (2)</li><li>• Constants.FV_MENU_CUSTOM (3)</li></ul>
HelpDir	string	Path of the FrameMaker product help directory
HomeDir	string	Path of \$FMHOME
UserSettingsDir	string	Name of the FrameMaker app directory. As defined by: %appdata%\Adobe\FrameMaker\<<version number>  Returns the HomeDir if the app directory is not found.
ImportFilters	string	List of import filters available in the current session.
InitDir	string	Directory path of \$FMHOME/fminit
FontAngleNames	Strings	List of font angles available in the current session.

Property name	Data Type	Description
FontFamilyAttributes	<a href="#">Ints</a>	<p>An array of flags that indicate attributes for each font family listed by FontFamilyNames. This array of integers is indexed the same as the list of font family names, and corresponds directly to that list. Each Int is a packed field; the high order 16 bits indicate a surrogate font, and the low order bits indicate attributes for the font family. The flags, their mask values, and their meaning follow:</p> <ul style="list-style-type: none"> <li>• Constants.FV_FAMILY_VISIBLE (0x00000001) - Family is visible in menu.</li> <li>• Constants.FV_FAMILY_SELECTABLE (0x00000002) - Family can be selected in menu.</li> <li>• Constants.FV_FAMILY_MAPPED (0x00000004) - Family is always mapped to another family.</li> <li>• Constants.FV_FAMILY_SURROGATE (0xFFFF0000): The family mapped to, if Constants.FV_FAMILY_MAPPED (0x00000004) is True.</li> </ul>
FontFamilyNames	<a href="#">Strings</a>	List of font family names available in the current session. This list does not include combined fonts.
FontVariationNames	<a href="#">Strings</a>	List of font variations available in the current session.
FontWeightNames	<a href="#">Strings</a>	List of font weights available in the current session.
Gravity	int	1 if Gravity is turned on for the session.
GreekSize	int	Size at which to greek text.
HostName	string	Name of the host computer.
IconBarOn	int	1 if the four icons that appear on the upper-right side of the document window are on. Changing this property affects only the documents that are opened subsequently; it does not affect documents that are already open.
IsIconified	int	1 if the FrameMaker product window is iconified.
IsInFront	int	1 if the FrameMaker product window is in front of other application windows. You can use this property to bring the FrameMaker product to the front or back.
IsOnScreen	int	1 if the FrameMaker product window is visible on the screen.
Label	string	The title in the FrameMaker product window title bar.

Property name	Data Type	Description
Language	int	Product language. The value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_LANG_NOLANGUAGE (0x00)</li> <li>• Constants.FV_LANG_ENGLISH (0x01)</li> <li>• Constants.FV_LANG_BRITISH (0x02)</li> <li>• Constants.FV_LANG_GERMAN (0x03)</li> <li>• Constants.FV_LANG_FRENCH (0x05)</li> <li>• Constants.FV_LANG_SPANISH (0x07)</li> <li>• Constants.FV_LANG_ITALIAN (0x09)</li> <li>• Constants.FV_LANG_SWEDISH (0x11)</li> <li>• Constants.FV_LANG_JAPANESE (0x12)</li> <li>• Constants.FV_LANG_TRADITIONAL_CHINESE (0x13)</li> <li>• Constants.FV_LANG_SIMPLIFIED_CHINESE (0x14)</li> <li>• Constants.FV_LANG_KOREAN (0x15)</li> </ul>
MarkerNames	Strings	List of standard marker types for the current session. For versions prior to 5.5, this property returned the list of all marker types for the current session. In version 5.5, marker types are assigned to the document; use the <code>MarkerTypeNames</code> property of <code>Constants.FO_Doc(4)</code> to get the full list of marker types.
OpenDir	string	Directory in which FrameMaker is installed.
Path	string	Path to search to start the FrameMaker product.
Platform	string	Name of the platform on which the current session is running.
ProductIsDemo	bool	1 if the current session is for a demo version of FrameMaker.
ProductIsStructured	bool	1 if FrameMaker is running in structured mode for the current session.
ProductName	string	The FrameMaker product name. The names for FrameMaker+SGML indicate FrameMaker running under the structured product interface. FrameViewer is retained for backward compatibility. Can be one of: <ul style="list-style-type: none"> <li>• FrameMaker</li> <li>• FrameMaker+SGML</li> <li>• FrameViewer</li> <li>• DemoMaker</li> <li>• DemoMaker+SGML</li> </ul>
ProductFamily	string	The FrameMaker product family name.: <ul style="list-style-type: none"> <li>• FrameMaker</li> <li>• FrameMaker XML Author</li> </ul>

Property name	Data Type	Description
ViewQuickAccessBar	int	1 if the QuickAccess bar is visible.
ViewFormattingBar	int	1 if the formatting bar is visible.
Reformatting	int	1 if reformatting is enabled.
RememberMissingFontNames	int	1 if Remember Missing Font Names is activated.
ScreenHeight	int	Height of the FrameMaker product window in pixels.
ScreenWidth	int	Width of the FrameMaker product window in pixels.
ScreenX	int	The offset of the FrameMaker product window in pixels from the left side of the screen. If you set a value that would result in the product window being off the screen, that value is ignored and the old value is retained.
ScreenY	int	The offset of the FrameMaker product window in pixels from the top of the screen. If you set a value that would result in the product window being off the screen, that value is ignored and the old value is retained.
SgmlDir	string	Directory path of \$FMHOME/structure/sgml.
Snap	int	1 if Snap is turned on for the session.
StructureDir	string	Directory path of \$FMHOME/structure.
TmpDir	string	Path of the temporary directory for internal FrameMaker product processes; the directory specified by the DOS \$TEMP environment variable.
UserLogin	string	User login name.
UserName	string	User name.
Validating	int	1 if validation is enabled (FrameMaker only).
VersionMajor	int	Frame version number (before the decimal).
VersionMinor	int	Frame version number (after the decimal).
WindowSystem	string	Name of window system that the FrameMaker product is running under.
XmlDir	string	Directory path of \$FMHOME/structure/xml.
DoNotGenerateErrorLog	int	When 1, the FrameMaker error log is not generated.
NoFlashInPDF	bool	When True, the Flash content in the PDF becomes static. When False, Flash objects get embedded in the PDF and are playable. Default is False.
No3DInPDF	bool	When True, the 3D objects in the generated PDF become 2-dimensional. When False, 3D objects get embedded in the PDF and do not become 2 dimensional. Default is False.
DontShowWelcomeScreen	bool	True when FrameMaker is set not to show the Welcome screen.
EnableAutoSpellCheck	bool	When True, Auto Spell Check is enabled.
AllowNewFileURL	bool	When True, allows usage of file:/ and file: URLs for referring to a file.

Property name	Data Type	Description
CharacterReferencesOnXMLExport	bool	When <code>True</code> , if there are leading/trailing whitespaces (spaces and tabs) in elements, save the whitespaces as character references to enable round-tripping. Else set to <code>False</code> . Default: <code>False</code>
DoNotExportInvalidXML	bool	When <code>True</code> , FrameMaker does not save the XML on disk if there are validation errors during XML export.
DoNotExportInvalidXML	bool	When <code>True</code> , FrameMaker does not save the XML on disk if there are validation errors during XML export.
SuppressXMLParserWarnings	bool	When <code>True</code> , the warning messages about duplicate declarations in DTD/Schema do not appear in the FrameMaker log.
RemoveExtraWhiteSpacesOnXMLImport	bool	When <code>True</code> , the extra while spaces are removed while importing XML.
NoMultiMediaInPDF	bool	When <code>True</code> , the Rich Media (audio and video) objects in the PDF become static. When <code>False</code> , Rich Media objects get embedded in the PDF and are playable. Default is <code>False</code> .

## app methods

[Close](#), [GetNamedBook](#), [GetNamedCommand](#), [GetNamedMenu](#), [GetNamedMenuItemSeparator](#), [GetNamedObject](#), [GetProps](#), [NewNamedBook](#), [NewNamedCommand](#), [NewNamedMenu](#), [NewNamedMenuItemSeparator](#), [NewNamedObject](#), [ObjectValid](#), [SetProps](#).

# Chapter 6: Function Summary

## AFrame

### Delete

#### Description

The `Delete()` method is used to delete an object from a document. When you delete an object that contains another object, all child objects are also deleted with the parent object. For example, if you delete a frame, all objects within the frame are also deleted.

The `Delete()` method does not take any arguments.

The method returns `FE_SUCCESS` on success, else returns one of the following values in `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadDelete (-17)	Specified object could not be deleted.
Constants.FE_BadOperation (-27)	Function call specified an illegal operation.
Constants.FE_BadParameter (-43)	Function call specified an invalid parameter.
Constants.FE_NotMenu (72)	Object is a menu item but the document to delete does not belong to the menu.

#### Returns

`int`

#### Syntax

`Delete()`

### GetProps

#### Description

The `GetProps()` method retrieves the complete property list for a specified object.

The `GetProps()` method takes no arguments.

The method returns a `PropVals` object that contains an array of property-value pairs, on success.

If the `GetProps()` method fails, it sets the `len` field of the returned structure to 0 and assigns one of the following values to `FA_errno`.

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid object.

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the specified operation.

**Returns**

[PropVals](#)

**Syntax**

GetProps ()

**SetProps**

**Description**

The `SetProps ()` method sets the properties of a specified object.

The method takes one argument - An array containing the values of the object to be set.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadName (-33)	Specified name is illegal.
Constants.FE_BadNewFrame (-14)	The method cannot move the specified object to this frame.
Constants.FE_BadNewGroup (-15)	The method cannot move the specified object to this graphic object group (FO_Group).
Constants.FE_BadNewSibling (-16)	Object cannot be made a sibling of the specified object.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadPropNum (-4)	Specified property number is invalid.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_CantSmooth (-38)	Object cannot be smoothed.
Constants.FE_DupName (-32)	Property cannot be set to this name because it is already used by another object.
Constants.FE_GenRuleAmbiguous (-55)	General rule in structured document is ambiguous.
Constants.FE_GenRuleConnectorExpected (-57)	General rule in structured document is missing a connector.
Constants.FE_GenRuleItemExpected (-51)	General rule in structured document is missing a rule item.
Constants.FE_GenRuleLeftBracketExpected (-53)	General rule in structured document is missing a left bracket.
Constants.FE_GenRuleMixedConnectors (-52)	General rule in structured document has mixed connectors.
Constants.FE_GenRuleRightBracketExpected (-54)	General rule in structured document is missing a right bracket.
Constants.FE_GenRuleSyntaxError (-56)	General rule in structured document has a syntax error.
Constants.FE_GroupSelect (-9)	The method cannot select or deselect an object in the specified group.

Error	Reason
Constants.FE_HiddenPage (-40)	The specified value must be a hidden page (FO_HiddenPage).
Constants.FE_InvContextSpec ()	The method encountered an invalid context specification in a FrameMaker document.
Constants.FE_NotBookComponent (-26)	The specified value must be a book component (FO_BookComponent).
Constants.FE_NotFrame (-12)	The specified value must be a frame.
Constants.FE_NotGraphic (-11)	The specified value must be a graphic object.
Constants.FE_NotGroup (-13)	The specified value must be a graphic object group (FO_Group).
Constants.FE_NotTextFrame (-39)	The specified value must be a text column (FO_TextFrame).
Constants.FE_NotTextObject (-41)	The specified object must be a text object, such as a paragraph (FO_Pgf) or a flow (FO_Flow).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.
Constants.FE_OutOfRange (-7)	Specified property value is out of the legal range for the specified property.
Constants.FE_PageFrame (-37)	The specified value must be a page frame object (FO_UnanchoredFrame).
Constants.FE_ReadOnly (-6)	Property is read-only.
Constants.FE_WithinFrame (-10)	The specified object must first be moved to a different frame.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support this operation.

**Returns**

Void

**Syntax**

SetProps (setVal)

**Parameters**

Parameter name	Data Type	Optional	Description
setVal	PropVals	No	The property list

**ObjectValid**

**Description**

Indicates whether an object is valid or not.

The method does not take any arguments.

The method returns 1 if the object is valid, else returns 0.

**Returns**

int

**Syntax**

ObjectValid()



## Arc

### Delete

#### Description

Deletes the Arc object.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

### GetProps

#### Description

The `GetProps()` method retrieves the properties of the Arc object.

See [SetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

#### Syntax

```
GetProps()
```

### SetProps

#### Description

The `SetProps()` method sets the properties of the Arc object.

See [SetProps](#) under the AFrame class for more information.

#### Returns

Void

#### Syntax

```
SetProps(setVal)
```

#### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the Arc object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## AttrCondExpr

### ApplyAttributeExpression

#### Description

`ApplyAttributeExpression()` is the method to apply the attribute expression to the document.

The method returns `FE_Success` on success.

#### Returns

int

#### Syntax

```
ApplyAttributeExpression()
```

## GetProps

### Description

The `GetProps()` method gets the properties of the specified attribute expression.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the specified attribute expression.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the attribute expression is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Attribute

### Attribute

### Description

Describes a single attribute.

### Returns

[Attribute](#)

### Syntax

```
Attribute(name, values, valflags, allow)
```

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	Attribute name.
values	Strings	No	Attribute values.
valflags	uint	No	Validation error flags.
allow	uint	No	Allow error as special case.

# AttributeDef

## AttributeDef

**Description**

Describes a single attribute function.

**Returns**

AttributeDef

**Syntax**

AttributeDef(name, required, flags, attrType, choices, defValues, rangeMin, rangeMax)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	Attribute name.
required	int	No	1 if the attribute is required; 0 otherwise.
flags	uint	No	One of read-only, hidden or neither.
attrType	int	No	The attribute type.
choices	Strings	No	The list of choices if attrType is set to Constants.FV_AT_CHOICES(2)
defValues	Strings	No	The default value, if attribute is not mandatory.
rangeMin	string	No	The minimum allowed value (if any)
rangeMax	string	No	The maximum allowed value (if any)

The flags parameter indicates whether the attribute is read-only, hidden or neither.

Flag	Meaning
Constants.FV_AF_READ_ONLY(0x0001)	The attribute value is read-only.
Constants.FV_AF_HIDDEN(0x0002)	The attribute value is hidden.
NULL	The attribute value is neither read-only nor hidden.

The attrType parameter can have one of the following values:

Value	Meaning
Constants.FV_AT_STRING(0)	Any arbitrary text string.
Constants.FV_AT_STRINGS(1)	One or more arbitrary text strings.
Constants.FV_AT_CHOICES(2)	A value from a list of choices.
Constants.FV_AT_INTEGER(3)	A signed whole number (optionally restricted to a range of values).
Constants.FV_AT_INTEGERS(4)	One or more integers (optionally restricted to a range of values).
Constants.FV_AT_REAL(5)	A real number (optionally restricted to a range of values).
Constants.FV_AT_REALS(6)	One or more real numbers (optionally restricted to a range of values).
Constants.FV_AT_UNIQUE_ID(7)	A string that uniquely identifies the element.
Constants.FV_AT_UNIQUE_IDREF(8)	A reference to a UniqueID attribute.
Constants.FV_AT_UNIQUE_IDREFS(9)	One or more references to a UniqueID attribute.

## AttributeDefs

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

[AttributeDefs](#)

#### Syntax

`concat (value)`

#### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">AttributeDe fs</a>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

### Returns

[AttributeDef](#)

### Syntax

pop()

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

push(value)

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">AttributeDef</a>	No	Value to push into the array.

## Attributes

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

## concat

### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

### Returns

[Attributes](#)

### Syntax

concat(value)

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Attributes</a>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

[Attribute](#)

### Syntax

```
pop()
```

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

```
push(value)
```

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Attribute</a>	No	Value to push into the array.

## AttributesEx

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

## concat

### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array as the result of concatenation, with the given values to the end of the original array.

### Returns

[AttributesEx](#)

### Syntax

```
concat(value)
```

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">AttributeEx</a> x	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

[AttributeEx](#)

### Syntax

```
pop()
```

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

```
push(value)
```

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">AttributeEx</a>	No	Value to push into the array.

# BodyPage

## Delete

### Description

Deletes the specified Body Page.

See [Delete](#) under the AFrame class for more information.

### Returns

int



### Syntax

Delete()

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified Body Page.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

GetProps()

## SetProps

### Description

The `SetProps()` method sets the properties of the specified Body Page.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

SetProps(setVal)

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the Body Page is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

ObjectValid()

## ApplyPageLayout

### Description

The `ApplyPageLayout ()` method applies the layout of one page to another page.

The method returns `FE_Success` on success, and sets `FA_errno` with one of the following values on error:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker version does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specified an invalid operation.
<code>Constants.FE_BadParameter (-43)</code>	The specified parameter has an invalid value.
<code>Constants.FE_SystemError (-42)</code>	System Error.

### Returns

`int`

### Syntax

```
ApplyPageLayout (srcPage)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>srcPage</code>	<code>PageObject</code>	No	The page to which the layout is to be applied.

## Book

## Compare

### Description

The `Compare ()` method compares the differences between two versions of files and stores the result in a `CompareRet` data object.

You can OR the following values into the flags parameter:

Flag	Meaning
<code>Constants.FF_CMP_SUMMARY_ONLY (0x01)</code>	Generate a summary document, but not a composite document.
<code>Constants.FF_CMP_CHANGE_BARS (0x02)</code>	Turn on change bars in the composite document.
<code>Constants.FF_CMP_HYPERLINKS (0x04)</code>	Put hypertext links in the summary document.
<code>Constants.FF_CMP_SUMKIT (0x08)</code>	Open the summary document.
<code>Constants.FF_CMP_COMPKIT (0x10)</code>	Open the composite document.

The returned `CompareRet` object contains the pointers to the summary and composite results documents.

**Function Summary**

The method returns the comparison results in the `CompareRet` object on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadCompare (-35)	Older and newer documents are not the same types of files.
Constants.FE_CompareTypes (-34)	One of the files is not a FrameMaker document or book or one file is a book and the other is a document.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the operation.

**Returns**

[CompareRet](#)

**Syntax**

```
Compare(newerDoc, flags, insertCondTag, deleteCondTag, replaceText, compareThreshold)
```

**Parameters**

Parameter name	Data Type	Optional	Description
<code>newerDoc</code>	<a href="#">Doc</a>	No	The newer version of the document or book.
<code>flags</code>	<code>int</code>	No	Bit flags that specify how to generate the summary and composite documents. Specify 0 for the default flags.
<code>insertCondTag</code>	<code>string</code>	No	Indicates the condition tag to apply to insertions shown in the composite document. To specify a blank insert condition tag, enter the value <code>null</code> .
<code>deleteCondTag</code>	<code>string</code>	No	Denotes the condition tag to apply to deletions shown in the composite document. To specify a blank delete condition tag, enter the value <code>null</code> .
<code>replaceText</code>	<code>string</code>	No	Text to appear in place of the deleted text. To not specify any replacement text, enter the value <code>null</code> .
<code>compareThreshold</code>	<code>int</code>	No	Threshold expresses the percentage of words that can change before paragraphs are considered not equal. If two paragraphs are equal, word differences between them are shown within a paragraph in the composite document.  If the <code>compareThreshold</code> property is set to 0, the <code>Compare()</code> method uses the default value of 75.  If the <code>compareThreshold</code> property is set to any other value, the <code>Compare()</code> method uses the setting in the FrameMaker configuration files.

**NewBookComponentInHierarchy****Description**

The `NewBookComponentInHierarchy()` method inserts a book component at a specified position in a FrameMaker structured book, and returns the component that was newly added, on success.

The method returns the element that was added, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadBookId (-66)	Invalid book.
Constants.FE_BadCompPath (-68)	Component name specified for compName is invalid.
Constants.FE_BadNew (-23)	The object cannot be created.
Constants.FE_BookUnStructured (-67)	The specified book is unstructured.

### Returns

[Element](#)

### Syntax

```
NewBookComponentInHierarchy(compName, elemLoc)
```

### Parameters

Parameter name	Data Type	Optional	Description
compName	string	No	The name of the component to add.
elemLoc	<a href="#">ElementLoc</a>	No	The position at which to add the new component.

## SilentPrintDoc

### Description

The `SilentPrintDoc()` method prints a document or a book using the default print settings. Default print settings are the settings that appear in the Print dialog box when the user attempts to print a document.

`SilentPrintDoc()` initializes the print page size and printer name if they do not have values.

On success, the method returns `FE_Success`. On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_SystemError (-42)	Could not open or close the printer file.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.

### Returns

int

### Syntax

```
SilentPrintDoc()
```

## SimpleGenerate

### Description

The `SimpleGenerate()` method generates files for a book.

The method performs the same operation as choosing `Update Book` from the book `Edit` menu.

You must set up the book and its generated files before calling `SimpleGenerate()`.

**Function Summary**

The method returns `FE_Success` on success, and assigns `FA_errno` with one of the following values on failure:

Error	Reason
Constants.FE_BadOperation (-27)	The book is not self-consistent (book generates data in one file that is source data for another generated file, or page count continually changes for this operation); there is a duplicate file in the book; all files in the book are generated files.
Constants.FE_BadParameter (-43)	Not a valid book.
Constants.FE_SystemError (-42)	Could not allocate memory, or could not open or save one of the files in the book.

**Returns**

`int`

**Syntax**

```
SimpleGenerate(interactive, makeVisible)
```

**Parameters**

Parameter name	Data Type	Optional	Description
<code>interactive</code>	<code>int</code>	No	Set the interactive property to 1 to display messages and warnings; set the property to 0 to suppress display of messages and warnings.
<code>makeVisible</code>	<code>int</code>	No	Specifies whether to display the generated files. Set to 1 to display the files, or to 0 to suppress displaying the files.

**SimpleImportFormats****Description**

The `SimpleImportFormats()` method imports formats from a document to a document or a book.

**Note:** If you import formats to a book, `SimpleImportFormats()` imports formats to each book component for which the `ImportFmtInclude` property is set to `True`.

You can OR the following values into the `formatFlags` parameter to specify the formats to import.

Flag	What to Import
Constants.FF_UFF_COLOR (0x0100)	Colors
Constants.FF_UFF_COMBINED_FONTS (0x0800)	Combined fonts
Constants.FF_UFF_COND (0x0010)	Conditions
Constants.FF_UFF_DOCUMENT_PROPS (0x0400)	Document properties
Constants.FF_UFF_FONT (0x0002)	Character Catalog formats
Constants.FF_UFF_MATH (0x0200)	Equation settings
Constants.FF_UFF_PAGE (0x0004)	Page layouts
Constants.FF_UFF_PGF (0x0001)	Paragraph Catalog formats
Constants.FF_UFF_REFPAGE (0x0020)	Reference pages
Constants.FF_UFF_REMOVE_EXCEPTIONS (0x8000)	Remove exception formats from target documents

Flag	What to Import
Constants.FF_UFF_REMOVE_PAGE_BREAKS (0x4000)	Remove all forced page breaks from target documents
Constants.FF_UFF_TABLE (0x0008)	Table Catalog formats
Constants.FF_UFF_VAR (0x0040)	Variable formats
Constants.FF_UFF_XREF (0x0080)	Cross-reference formats

The method returns `FE_Success` on success. If the method fails, it sets `FA_errno` with one of the following values:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support books.
Constants.FE_BadDocId (-2)	Invalid book or document.
Constants.FE_Canceled (-44)	User canceled the operation.
Constants.FE_FailedState (-45)	FrameMaker failed to open one or more of the book's document files during the import operation.

### Returns

`int`

### Syntax

`SimpleImportFormats (fromDoc, formatFlags)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>fromDoc</code>	<code>Doc</code>	No	The document from which the formats are to be imported.
<code>formatFlags</code>	<code>int</code>	No	Bit field specifying the formats to be imported. Specify 0 for the default flags.

## SimpleImportElementDefs

### Description

The `SimpleImportElementDefs ()` method imports element definitions and the format change list catalog from an EDD or FrameMaker document or book to another FrameMaker document or book.

You can OR the following flags into the `importFlags` parameter:

Flag	Meaning
Constants.FF_IED_REMOVE_OVERRIDES (0x0001)	Clear format overrides.
Constants.FF_IED_REMOVE_BOOK_INFO (0x0002)	Clears formatting inherited from a parent book, if the element definitions are being imported from a document.
Constants.FF_IED_DO_NOT_IMPORT_EDD (0x0004)	If the source document is an EDD, setting this value does NOT treat it as an EDD; instead the method just imports its element catalog.

**Function Summary**

Flag	Meaning
Constants.FF_IED_NO_NOTIFY (0x0008)	Do not issue the Constants.FA_Note_PreImportElemDefs (64) or Constants.FA_Note_PostImportElemDefs (65) notifications.

The method returns `FE_Success` on success. If the method fails, it assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current product version is not Structured FrameMaker.
Constants.FE_BadDocId (-2)	Invalid book or document.

**Returns**

`int`

**Syntax**

```
SimpleImportElementDefs (fromDocOrBook, importFlags)
```

**Parameters**

Parameter name	Data Type	Optional	Description
fromDocOrBook	<code>Doc   Book</code>	No	The document or book from which to import element definitions.
importFlags	<code>int</code>	No	Specifies how to import formats.

## GetUniqueObject

**Description**

The `GetUniqueObject ()` method retrieves the object from its persistent unique identifier (UID). The FrameMaker product assigns a UID to each object in a document or book that is not identified by a name. The UID, unlike the object's ID, does not change from one session to another.

The `GetUniqueObject()` method returns the object on success. On failure, it sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document or book.
Constants.FE_TypeUnNamed (-19)	Objects of the specified type are not identified by UIDs.
Constants.FE_NameNotFound (-20)	Object with specified UID could not be found.

**Returns**

`Object`

**Syntax**

```
GetUniqueObject (objType, unique)
```

**Parameters**

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object (for example, FO_Pgf).
unique	int	No	The unique ID of the object.

**Close**

**Description**

Closes a document, book, dialog box, or Frame session.

The `flags` parameter specifies whether to abort or to close open documents or books if they have unsaved changes. Set the `Constants.FF_CLOSE_MODIFIED` flag to `True (1)` to close open documents and books regardless of their state.

The method sets `FE_Success` on success. On failure, the method assigns the following value to `FA_errno`:

Error	Reason
<code>Constants.FE_DocModified (-8)</code>	The document was modified and the <code>flags</code> parameter was set to 0.

**Returns**

int

**Syntax**

`Close(flags)`

**Parameters**

Parameter name	Data Type	Optional	Description
flags	int	No	The <code>Close()</code> method closes the document or book.  <i>Note: Flags specify whether to abort or to close open documents or books if they have unsaved changes. Set the <code>FF_CLOSE_MODIFIED</code> flag to <code>True (1)</code> to close open documents and books regardless of their state.</i>

**SimpleSave**

**Description**

The `SimpleSave()` method saves a book.

If you set the `interactive` parameter to `False` and you specify the book's current name for `saveAsName`, FrameMaker saves the book under its current name.

If you specify another filename for `saveAsName`, FrameMaker saves the book to that filename.

If you specify an empty string (`""`), FrameMaker does not save the file. Instead it sets `FA_errno` to `Constants.FE_BadParameter (-43)`.



If you set the `interactive` parameter to `True`, FrameMaker displays the Save dialog box and allows the user to choose a filename. The book's current name appears as the default file name.

The `SimpleSave()` method returns the book it saved, on success.

On failure, the method sets `FA_errno`, to one of the following values:

Error	Reason
<ul style="list-style-type: none"> <li>Constants.FE_Canceled (-44)</li> <li>Constants.FE_CanceledByClient (-86)</li> </ul>	<ul style="list-style-type: none"> <li>Constants.FV_FileNotWritable (33) - File was not writable.</li> <li>Constants.FV_BadSaveFileName (34) - The specified file name is not allowed by the operating system.</li> <li>Constants.FV_BadFileId (35) - The file's operating system ID was bad.</li> <li>Constants.FV_CancelSaveFilesInUse (50) - The file is in use and the user did not or could not reset the lock. Or the file is in use, and the <code>Constants.FS_FileIsInUse (17)</code> parameter is set to <code>Constants.FV_DoCancel (0)</code>, or it is set to <code>Constants.FV_ResetLockAndContinue (7)</code> but FrameMaker could not reset the lock.</li> <li>Constants.FV_CancelSaveModDateChanged (52) - The file has changed since the last time it was opened or saved in the current session.</li> <li>Constants.FV_LockCouldntBeReset (2) - File lock could not be reset.</li> <li>Constants.FV_LockWasReset (0) - File lock was reset.</li> <li>Constants.FV_LockNotReset (1) - File lock was not reset.</li> <li>Constants.FV_FilesViewOnly (4) - File was View Only.</li> <li>Constants.FV_NonPortableSaveName (37) - File had a name that is not portable (Macintosh only).</li> <li>Constants.FV_NonPortableFileRefs (38) - Referenced graphics files had filenames that are not portable (Macintosh only).</li> </ul>
Constants.FE_WrongProduct (-60)	The specified file contains structure and the current product interface is not Structured FrameMaker.
Constants.FE_FailedState (-45)	The filename was invalid.
Constants.FE_BadParameter (-43)	
Constants.FE_FilterFailed (-89)	Constants.FV_InvalidSaveFilter (42) - The filter specified by <code>Constants.FS_SaveFileTypeHint (16)</code> is not installed, or the syntax for <code>Constants.FS_SaveFileTypeHint (16)</code> is invalid.

### Returns

[Book](#)

### Syntax

`SimpleSave(saveAsName, interactive)`

### Parameters

Parameter name	Data Type	Optional	Description
saveAsName	string	No	The absolute path in which to save the document or book.
interactive	int	No	Specifies whether FrameMaker displays messages and warnings to the user. Set this parameter to 1 to display messages and warnings.

## NewSeriesObject

### Description

The `NewSeriesObject ()` method creates a series object. Series objects include the following object types:

- BodyPage
- BookComponent
- Pgf

The method allows you to specify the position in the series at which to add the new object.

The method returns the location of the new object, on success. If the method fails, it sets `FA_erno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid book.
Constants.FE_BadNew (-23)	Object cannot be created.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_NotBodyPage (-24)	The <code>previous</code> parameter must specify the location of a body page.
Constants.FE_NotPgf (-25)	The <code>previous</code> parameter must specify the location of a paragraph.
Constants.FE_NotBookComponent (-26)	The previous parameter must specify the location of a book component.

### Returns

Object

### Syntax

`NewSeriesObject (objType, previous)`

### Parameters

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object to create (for example, <code>FO_BodyPage</code> or <code>FO_Pgf</code> ).
previous	Object	No	The location of the object in the series after which to add the new object. To add a paragraph at the start of a flow, specify the location of the flow. To add an object at the beginning of any other series, specify 0.

## NewSeriesBookComponent

### Description

The `NewSeriesBookComponent ()` method creates a Book Component.

The method allows you to specify the position in the series at which to add the new object.

See the [NewSeriesObject](#) method for more information.

**Function Summary****Returns**[BookComponent](#)**Syntax**`NewSeriesBookComponent (previous)`**Parameters**

Parameter name	Data Type	Optional	Description
previous	Object	No	The location of the object in the series after which to add the new Book Component. To add a paragraph at the start of a flow, specify the location of the flow. To add a Book Component at the beginning of any other series, specify 0.

## NewBookComponentOfTypeInHierarchy

**Description**

TBD

**Returns**[BookComponent](#)**Syntax**`NewBookComponentOfTypeInHierarchy (compName, compType, elemLoc)`**Parameters**

Parameter name	Data Type	Optional	Description
compName	string	No	TBD
compType	int	No	TBD
elemLoc	<a href="#">ElementLoc</a>	No	TBD

## Save

**Description**

The `Save ()` method saves a book. The method allows you to script the way FrameMaker saves the file and to specify responses to warnings and messages that arise while the file is being saved.

You can save a file under its current name or save it as a new file.

The property list for the `saveParams` parameter is as follows:

Property	Meaning
Constants.FS_SavedFileName (1)	A string that specifies the full path of the saved file.
Constants.FS_SaveNativeError (2)	The error condition. If the file is saved successfully, this property is set to <code>FE_Success</code> .
Constants.FS_SaveStatus (3)	A bit field indicating what happened when the file was saved.

The method returns the document it saved, on success.

On failure the method sets `FA_errno` to one of the following values:

Error	Reason
<ul style="list-style-type: none"> <li>• Constants.FE_Canceled (-44)</li> <li>• Constants.FE_CanceledByClient (-86)</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FV_FileNotWritable (33) - File was not writable.</li> <li>• Constants.FV_BadSaveFileName (34) - The specified filename is not allowed by the operating system.</li> <li>• Constants.FV_BadFileId (35) - The file's operating system ID was bad.</li> <li>• Constants.FV_CancelSaveFilesInUse (50) - The file is in use and the user did not or could not reset the lock. Or the file is in use, and the <code>Constants.FS_FileIsInUse (17)</code> parameter is set to <code>Constants.FV_DoCancel (0)</code>, or it is set to <code>Constants.FV_ResetLockAndContinue (7)</code> but FrameMaker could not reset the lock.</li> <li>• Constants.FV_CancelSaveModDateChanged (52) - The file has changed since the last time it was opened or saved in the current session.</li> <li>• Constants.FV_LockCouldntBeReset (2) - File lock could not be reset.</li> <li>• Constants.FV_LockWasReset (0) - File lock was reset.</li> <li>• Constants.FV_LockNotReset (1) - File lock was not reset.</li> <li>• Constants.FV_FilesViewOnly (4) - File was View Only.</li> <li>• Constants.FV_NonPortableSaveName (37) - File had a name that is not portable (Macintosh only).</li> <li>• Constants.FV_NonPortableFileRefs (38) - Referenced graphics files had filenames that are not portable (Macintosh only).</li> </ul>
<p>Constants.FE_WrongProduct (-60)</p>	<p>The specified file contains structure and the current product interface is not Structured FrameMaker.</p>
<p>Constants.FE_FailedState (-45)            Constants.FE_BadParameter (-43)</p>	<p>The filename was invalid.</p>
<p>Constants.FE_FilterFailed (-89)</p>	<p>Constants.FV_InvalidSaveFilter (42) - The filter specified by <code>Constants.FS_SaveFileTypeHint (16)</code> is not installed, or the syntax for <code>Constants.FS_SaveFileTypeHint (16)</code> is invalid.</p>

**Returns**

Object

**Syntax**

`Save (saveAsName, saveParams, saveReturnParams)`

**Parameters**

Parameter name	Data Type	Optional	Description
saveAsName	string	No	The path for saving the document or book.
saveParams	PropVals	No	A property list that tells FrameMaker how to save the file and how to respond to errors and other conditions. Use GetSaveDefaultParams() or AllocatePropVals() to create and allocate memory for this property list. To use the default list, specify null.
saveReturnParams	PropVals	No	A property list that returns information about how the FrameMaker product saved the file.

**UpdateBook**

**Description**

The UpdateBook () method updates a book.

The method allows you to specify a script (property list) specifying how to update the book and how to deal with error and warning conditions.

For example, you can specify whether to abort or to continue updating a book if it contains view-only documents.

*Note:* Always initialize the pointer to the property list that you specify for updateReturnParamspp to null before you call UpdateBook().

To get a property list to specify for the updateParams parameter, use GetUpdateBookDefaultParams () or create the list from scratch.

The following are the properties that you can use in the property list:

Property	Meaning
Constants.FS_AlertUserAboutFailure (2)	Alert user with warnings and messages if necessary. Set this parameter to True or False as desired. <ul style="list-style-type: none"> <li>• True - Notify user when unexpected conditions occur.</li> <li>• False - Do not notify user when unexpected conditions occur.</li> </ul>
Constants.FS_AllowInconsistentNumProps (6)	Allow the FrameMaker product to update numbering, text insets and other properties of all the FrameMaker documents in the book, even if there are documents in the book with numbering properties that do not match the properties specified in the book. The permissible values are as follows: <ul style="list-style-type: none"> <li>• Constants.FV_DoOK (1) - Update numbering even if there are inconsistent properties in the book.</li> <li>• Constants.FV_DoCancel (0) - Cancel the update operation when FrameMaker encounters a document with inconsistent numbering properties.</li> <li>• Constants.FV_DoShowDialog (4) - Show a dialog box and let the user decide.</li> </ul>

Property	Meaning
Constants.FS_AllowNonFMFiles (1)	<p>Allow the FrameMaker product to update numbering, text insets, and other properties of all the FrameMaker documents in the book, even if there are documents in the book that were not created by FrameMaker. The permissible values are as follows:</p> <ul style="list-style-type: none"> <li>• Constants.FV_DoOK (1) - Update the book even if the book contains files not created by FrameMaker.</li> <li>• Constants.FV_DoCancel (0) - Cancel the update operation when it encounters a document not created by FrameMaker.</li> <li>• Constants.FV_DoShowDialog (4) - Show a dialog box and let the user decide.</li> </ul>
Constants.FS_AllowViewOnlyFiles (3)	<p>Allow the FrameMaker product to update view-only documents in the book. The permissible values are as follows:</p> <ul style="list-style-type: none"> <li>• Constants.FV_DoOK (1) - Update the view-only documents.</li> <li>• Constants.FV_DoCancel (0) - Cancel the entire update operation when it encounters a view-only document.</li> <li>• Constants.FV_DoShowDialog (4) - Show a dialog box and let the user decide.</li> </ul>
Constants.FS_MakeVisible (5)	<p>Make newly generated files (lists and indexes) visible. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>• True - Make visible.</li> <li>• False - Do not make visible.</li> </ul>
Constants.FS_ShowBookErrorLog (4)	<p>Display the book error log for this update operation. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>• False - Do not display the error log; all warnings and errors are redirected to the console.</li> <li>• True - Display the error log.</li> </ul>
Constants.FS_UpdateBookGeneratedFiles (7)	<p>Update generated files such as TOC, lists, and indexes. Only update those generated files that have GenerateInclude set to True in their associated FO_BookComponent objects. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>• True - Update generated files.</li> <li>• False - Do not update generated files.</li> </ul>
Constants.FS_UpdateBookMasterPages (12)	<p>Update automatic assignment of master pages in all the documents of the book. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>• True - Update master pages.</li> <li>• False - Do not update master pages.</li> </ul>
Constants.FS_UpdateBookNumbering (8)	<p>Update numbering in all the documents of the book. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>• True - Update numbering.</li> <li>• False - Do not update numbering.</li> </ul>
Constants.FS_UpdateBookOleLinks (9)	<p>Update OLE links in all the documents of the book. Takes the value True or False.</p> <ul style="list-style-type: none"> <li>• True - Update OLE links.</li> <li>• False - Do not update OLE links.</li> </ul>

Property	Meaning
Constants.FS_UpdateBookTextReferences (10)	Update text insets in all the documents of the book. Takes the value True or False. <ul style="list-style-type: none"> <li>• True - Update text insets.</li> <li>• False - Do not update text insets.</li> </ul>
Constants.FS_UpdateBookXRefs (11)	Update cross-references in all the documents of the book. Takes the value True or False. <ul style="list-style-type: none"> <li>• True - Update cross-references.</li> <li>• False - Do not update cross-references.</li> </ul>

The property list to which `updateReturnParams` is set, has the following property:

Property	Meaning
Constants.FS_UpdateBookStatus (1)	A bit field to indicate what happened during the update.

To determine if a particular `Constants.FS_UpdateBookStatus (1)` bit is set, use `CheckStatus ()`.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid book
Constants.FE_BadOperation ()	<ul style="list-style-type: none"> <li>• Constants.FV_BookNotSelfConsistent (0) - The book is not self-consistent (book generates data in one file that is source data for another generated file, or page count continually changes for this operation).</li> <li>• Constants.FV_DuplicateFileInBook (1) - One or more files in the book is a duplicate of another file.</li> <li>• Constants.FV_NoNonGeneratedFilesInBook (2) - The only files in the book are generated files.</li> </ul>
Constants.FE_BadParameter (-43)	<ul style="list-style-type: none"> <li>• Constants.FV_BadUpdateBookFileId (64) - Specified book is invalid.</li> <li>• Constants.FV_BadUpdateBookScriptValue (65) - The update book script contains an invalid property value.</li> </ul>
Constants.FE_Canceled (-44) Constants.FE_CanceledByClient (-86)	<ul style="list-style-type: none"> <li>• Constants.FV_CancelInconsistentNumPropsInFileInBook (34) - One or more of the book's document files has numbering properties that are inconsistent with the properties stored in the book.</li> <li>• Constants.FV_CancelNonFMFileInBook (32) - One or more of the book's document files is not a FrameMaker file.</li> <li>• Constants.FV_CancelViewOnlyFileInBook (33) - One or more of the book's document files is view-only.</li> <li>• Constants.FV_UserCanceledUpdateBook (35) - The user canceled the update operation.</li> </ul>
Constants.FE_SystemError (-42)	<ul style="list-style-type: none"> <li>• Constants.FV_FileInBookNotOpened (96) - One or more files in the book could not be opened.</li> <li>• Constants.FV_FileInBookNotSaved (97) - One or more files in the book could not be saved.</li> <li>• Constants.FV_TooManyWindowsUpdateBook (98) - Not enough memory. Too many windows were open.</li> </ul>

**Returns**

`int`

**Syntax**

UpdateBook(updateParams, updateReturnParams)

**Parameters**

Parameter name	Data Type	Optional	Description
updateParams	PropVals	No	A property list specifying how to update the book and how to respond to errors and other conditions. To use the default list, specify null.
updateReturnParams	PropVals	No	A property list that provides information about how the FrameMaker product updated the book. This parameter must be initialized before you call UpdateBook().

**Import**

**Description**

The Import() method imports text or graphics into a document.

Import() allows you to specify a script (property list) specifying how to import text or graphics and how to deal with error and warning conditions.

For example, you can specify whether to import a file by reference or by copy.

If you import a file by reference, Import() creates an inset. The following list summarizes the types of files you can import with Import() and the types of inset objects it creates when you import them by reference:

- Graphics - FO\_Inset
- Text - FO\_TiText, FO\_TiTextTable
- Frame binary document - FO\_TiFlow
- MIF - FO\_TiFlow

**Note:** When importing a graphic, you can specify that it be imported at its default resolution by setting the Constants.FS\_GraphicDpi (15) property to 0 and setting the Constants.FS\_FitGraphicInSelectedRect (14) property to False. If the graphic has no default resolution, it is imported at 72 dpi

To get a property list to specify for the importParams parameter, use GetImportDefaultParams() or create the list from scratch.

**Note:** Always initialize the pointer to the property list that you specify for importReturnParams to null before you call Import().

The following are the list of properties that can be imported:

Property	Meaning
Constants.FS_AlertUserAboutFailure (2)	Alert user if an unexpected condition, such as an unrecognized file type, occurs. Set to either True or False <ul style="list-style-type: none"> <li>• True - Notify user when unexpected conditions occur.</li> <li>• False - Do not notify user when unexpected conditions occur.</li> </ul>



Property	Meaning
Constants.FS_DisallowDoc (8)	<p>Disallow importing FrameMaker binary documents. Set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Do not allow importing FrameMaker binary documents.</li> <li>• False - Allow importing FrameMaker binary documents.</li> </ul>
Constants.FS_DisallowFilterTypes (11)	<p>Disallow importing filterable files. Set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Do not allow importing filterable files.</li> <li>• False - Allow importing filterable files.</li> </ul>
Constants.FS_DisallowMIF (7)	<p>Disallow importing MIF files. Set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Do not allow importing MIF files.</li> <li>• False - Allow importing MIF files.</li> </ul>
Constants.FS_DisallowGraphicTypes (9)	<p>Disallow importing graphic files. Set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Do not allow importing graphic files.</li> <li>• False - Allow importing graphic files.</li> </ul>
Constants.FS_DisallowMacEditions (10)	<p>Disallow importing Macintosh editions of files. Set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Do not allow importing Macintosh editions of files.</li> <li>• False - Allow importing Macintosh editions of files.</li> </ul>
Constants.FS_DisallowPlainText (12)	<p>Disallow importing Text Only files. Set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Do not allow importing Text Only files.</li> <li>• False - Allow importing Text Only files.</li> </ul>
Constants.FS_DisallowSgml (33)	<p>Disallow importing SGML documents. Set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Do not allow importing SGML documents.</li> <li>• False - Allow importing SGML documents.</li> </ul>
Constants.FS_DisallowXml (43)	<p>Disallow importing XML documents. Set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Do not allow importing XML documents.</li> <li>• False - Allow importing XML documents.</li> </ul>
Constants.FS_DontNotifyAPIClients (3)	<p>Notify other clients of the import operation. Set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Do not notify other clients of the import operation.</li> <li>• False - Notify other clients of the import operation.</li> </ul>
Constants.FS_FileTypeHint (34)	<p>If the file is filterable, this property is a string that enables FrameMaker to automatically call the correct filter to filter the file.</p> <p>If the file is not filterable, this property contains the value <code>null</code>.</p>
Constants.FS_FilesSgmlDoc (32)	<p>File is an SGML document. The possible values are:</p> <ul style="list-style-type: none"> <li>• Constants.FV_DoOK (1) - Import it anyway.</li> <li>• Constants.FV_DoCancel (0) - Cancel the import operation.</li> <li>• Constants.FV_DoShowDialog (4) - Show a dialog box and let the user decide.</li> </ul>

Property	Meaning
Constants.FS_FilesXmlDoc (37)	File is an XML document. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_DoOK (1) - Import it anyway.</li> <li>• Constants.FV_DoCancel (0) - Cancel the import operation.</li> <li>• Constants.FV_DoShowDialog (4) - Show a dialog box and let the user decide.</li> </ul>
Constants.FS_ForceImportAsText (17)	Import the file as a Text Only document, even if it is a MIF file or a filterable file. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Import it as a Text Only file.</li> <li>• False - Import it in a format based on its type.</li> </ul>
Constants.FS_HowToImport (4)	Import file by reference or copy. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_DoByRef (5) - Import file by reference.</li> <li>• Constants.FV_DoByCopy (6) - Import file by copy.</li> <li>• Constants.FV_DoUserChoice (7) - Allow user to choose how to import the file.</li> </ul>
Constants.FS_ImportAsType (36)	Specify the format of the file to import. Possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_AUTORECOGNIZE (0x00) - Default value; recognize the file type automatically.</li> <li>• Constants.FV_TYPE_BINARY (0x01) - A FrameMaker binary file.</li> <li>• Constants.FV_TYPE_MIF (0x02) - A MIF file.</li> <li>• Constants.FV_TYPE_TEXT (0x03) - A file containing plain text.</li> <li>• Constants.FV_TYPE_SGML (0x04) - An SGML file.</li> <li>• Constants.FV_TYPE_XML (0x05) - An XML file.</li> <li>• Constants.FV_TYPE_FILTER (0xFF) - Use a filter to import this file. You must specify a valid file type hint for the Constants.FS_FileTypeHint (34) property.</li> </ul>
Constants.FS_ManualUpdate (5)	Update inset manually. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Update inset automatically.</li> <li>• False - Do not update inset manually.</li> </ul>
Constants.FS_SgmlImportApplication (35)	Retained for compatibility with older versions of FrameMaker. Use the Constants.FS_StructuredImportApplication (35) property instead.
Constants.FS_StructuredImportApplication (35)	String specifying the name of the structure application to use when importing an XML or SGML file. This property takes precedence over any other structure application specification. If the specified application does not exist, the calling function will fail.  If this property is not set, then the value of this property is <code>null</code> .
Constants.FS_ShowBrowser (1)	Display the Import dialog box. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Display the Import dialog box.</li> <li>• False - Do not display the Import dialog box.</li> </ul>
Constants.FS_TextInsetName (6)	Denotes the Inset name. If not set, the value of this property is <code>null</code> .

The syntax of the string passed to the `Constants.FS_FileTypeHint (34)` is as follows:

**Function Summary**

```
record_vers vendor format_id platform filter_ers filter_name
```

**Note:** The fields in the record are not separated by spaces. Spaces are significant data except those that appear in the `filter_name` field.

For example:

```
0001PGRFPICMAC61.0 Built-in PICT reader0001ORBLSVGIA11 Platform SVG reader for 7.0
```

Each field of the record (except `filter_name`) specifies a four-byte code. If a code contains fewer than four alphanumeric characters, the remaining bytes must be filled out with spaces.

`record_ers` specifies the version of the record, currently 0001.

`vendor` is a code specifying the filter's vendor. The code is a string of four characters. The following table lists the possible codes

Code	Meaning
PGRF	Built-in Frame filters.
FAPI	External Frame FDK client filter.
FFLT	External Frame filters.
IMAG	External ImageMark filters.
XTND	External XTND filters.

`format_id` is a code specifying the format that the filter translates. The code is a string of four characters. The following are the possible codes:

Code	Meaning
CDR	CorelDRAW
CGM	Computer Graphics Metafile
DIB	Device-independent bitmap (Windows)
DRW	Micrografx CAD
DXF	Autodesk Drawing eXchange file (CAD files)
EMF	Enhanced Metafile (Windows)
EPSB	Encapsulated PostScript Binary (Windows)
EPSD	Encapsulated PostScript with Desktop Control Separations (DCS)
EPSF	Encapsulated PostScript (Macintosh)
EPSI	Encapsulated PostScript Interchange
FRMI	FrameImage
FRMV	FrameVector
G4IM	CCITT Group 4 to Image
GEM	GEM file (Windows)
GIF	Graphics Interchange Format (CompuServe)
HPGL	Hewlett-Packard Graphics Language

Code	Meaning
IGES	Initial Graphics Exchange Specification (CAD files)
IMG4	Image to CCITT Group 4 (UNIX)
MooV	QuickTime Movie
OLE	Object Linking and Embedding Client (Microsoft)
PCX	PC Paintbrush
PICT	QuickDraw PICT
PNTG	MacPaint
SNRF	Sun Raster File
SRGB	SGI RGB
SVGI	Scalable Vector Graphics
TIFF	Tag Image File Format
WMF	Windows Metafile
WPG	WordPerfect Graphics
XWD	X Windows System Window Dump file

`platform` is a code specifying the platform on which the filter was run. The code is a string of four characters. The following are the possible codes:

Code	Meaning
MAC6	Macintosh 68000 series
MACP	Power Macintosh
WINT	Windows NT
WIN3	Windows 3.1
WIN4	Windows 95
OS/2	IBM OS/2
UNIX	Generic X/11 (Sun, HP)

`filter_vers` is a string of four characters identifying the version of the filter on that platform. For example, version 1.0 of a filter is represented by the string 1.0.

`filter_name` is a text string (up to 31 characters long) that describes the filter.

`Import()` uses the following properties only for importing FrameMaker documents and MIF files:

Property	Meaning
<code>Constants.FS_FilesMakerDoc (18)</code>	File is a FrameMaker binary document or a MIF file. The possible values are: <ul style="list-style-type: none"> <li><code>Constants.FV_DoOK (1)</code> - Import it anyway.</li> <li><code>Constants.FV_DoCancel (0)</code> - Cancel the import operation.</li> <li><code>Constants.FV_DoShowDialog (4)</code> - Show a dialog box and let the user decide.</li> </ul>

Property	Meaning
Constants.FS_FormatImportedText (22)	Format the imported text. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_EnclosingDoc (1) - Use formatting in the enclosing document.</li> <li>• Constants.FV_PlainText (2) - Format the imported text as plain text.</li> <li>• Constants.FV_SourceDoc (0) - Use formatting from the source documents.</li> </ul>
Constants.FS_ImportFlowPageSpace (21)	If Constants.FS_UseMainFlow (19) is False, then Constants.FS_ImportFlowPageSpace (21) indicates the type of pages to search for the flow specified by the Constants.FS_ImportFlowTag (20) property. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_BodyPage (10) - Search body pages.</li> <li>• Constants.FV_ReferencePage (9) - Search reference pages.</li> </ul>
Constants.FS_ImportFlowTag (20)	If Constants.FS_UseMainFlow (19) is False, then Constants.FS_ImportFlowTag (20) indicates the name of the flow to import. If this value is not set, the default value is null.
Constants.FS_RemoveManualPageBreaks (23)	Remove manual page breaks if FS_FormatImportedText is set to Constants.FV_EnclosingDoc (1). Set to either True or False. <ul style="list-style-type: none"> <li>• True - Remove manual page breaks.</li> <li>• False - Do not remove manual page breaks.</li> </ul>
Constants.FS_RemoveOverrides (24)	Remove format overrides if Constants.FS_FormatImportedText (TBD) is set to Constants.FV_EnclosingDoc (1). Set to either True or False. <ul style="list-style-type: none"> <li>• True - Remove format overrides.</li> <li>• False - Do not remove format overrides.</li> </ul>
Constants.FS_UseMainFlow (19)	Import text from specified document's main flow. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Import the text from the main flow.</li> <li>• False - Do not import the text from the main flow.</li> </ul>

Import () uses the following properties only for importing graphics files:

Property	Meaning
Constants.FS_FileIsGraphic (13)	File is a graphic file. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_DoOK (1) - Import it.</li> <li>• Constants.FV_DoCancel (0) - Cancel import operation.</li> <li>• Constants.FV_DoShowDialog (4) - Display a dialog box and let the user decide.</li> </ul>
Constants.FS_FitGraphicInSelectedRect (14)	Fit the graphic in the selected graphic frame. Set to either True or False. <ul style="list-style-type: none"> <li>• True - Fit the graphic in the frame.</li> <li>• False - Do not fit the graphic in the frame.</li> </ul>
Constants.FS_GraphicDpi (15)	Integer specifying dots per inch (DPI) at which to import the graphic. The default value is 72 (to specify 72 DPI).
Constants.FS_InsetRasterDpi (TBD)	Denotes the read-only value to indicate the DPI of the bitmap facet when importing SVG data.
Constants.FS_RasterDpi (38)	Specifies the DPI for the bitmap facet when importing SVG data. This corresponds to any of the DPI settings in the Imported Graphic Scaling dialog box.

Property	Meaning
Constants.FS_RasterImageHeight (40)	Specifies the height of the rectangle in which to import the graphic, when importing SVG data.  <i>Note: You should always specify this value in points. This property corresponds to the height for the Fit in Selected Rectangle setting in the Imported Graphic Scaling dialog box.</i>
Constants.FS_RasterImageWidth (39)	Specifies the width of the rectangle in which to import the graphic, when importing SVG data.  <i>Note: You should always specify this value in points. This property corresponds to the width for the Fit in Selected Rectangle setting in the Imported Graphic Scaling dialog box.</i>
Constants.FS_ShowRasterDpiDialog (41)	For importing SVG data, show the Imported Graphic Scaling dialog box. Default is 1 (one). The values are: <ul style="list-style-type: none"> <li>• 1 (one) or any non-zero value: Display the Imported Graphic Scaling dialog box.</li> <li>• 0 (zero): Do not display the Imported Graphic Scaling dialog box.</li> </ul>

Import () uses the following properties only for importing ASCII text files:

Property	Meaning
Constants.FS_CellSeparator (29)	If Constants.FS_FilesText (16) is set to Constants.FV_DoImportAsTable (8), then this property denotes the delimiter or separator used to parse the text into cells. The default value is null.
Constants.FS_FilesText (16)	File is a Text Only file. The possible values are: <ul style="list-style-type: none"> <li>• Constants.FV_TextFile_EOLisEOP (12) - Import the file and convert each end-of-line into a paragraph break.</li> <li>• Constants.FV_TextFile_EOLisNotEOP (13) - Import the file but do not convert each end-of-line into a paragraph break.</li> <li>• Constants.FV_DoImportAsTable (8) - Import the file into a table.</li> <li>• Constants.FV_DoCancel (0) - Cancel the import operation.</li> </ul>
Constants.FS_ImportTblTag (25)	Specifies the table format to use, if Constants.FS_FilesText (16) is set to Constants.FV_DoImportAsTable (8). The default value is null.
Constants.FS_LeaveHeadingRowsEmpty (27)	Leave the heading rows empty if Constants.FS_FilesText (16) is set to Constants.FV_DoImportAsTable (8). Constants.FS_LeaveHeadingRowsEmpty (27) is set to either True or False. <ul style="list-style-type: none"> <li>• True - Leave heading rows empty.</li> <li>• False - Do not leave heading rows empty.</li> </ul>
Constants.FS_NumCellSeparators (30)	Denotes the number of spaces to use as a separator, if Constants.FS_FilesText (16) is set to Constants.FV_DoImportAsTable (8), and Constants.FS_CellSeparator (29) is set to a space (' '). The default value is 1.
Constants.FS_NumColumns (31)	Specifies the number of columns in the table, if Constants.FS_FilesText (16) is set to Constants.FV_DoImportAsTable (8), and Constants.FS_TreatParaAsRow (28) is False. The default value is 1.
Constants.FS_TblNumHeadingRows (26)	Specifies the number of heading rows in the table, if Constants.FS_FilesText (16) is set to Constants.FV_DoImportAsTable (8). The default value is 1.

Property	Meaning
Constants.FS_TreatParaAsRow (28)	<p>Convert each line in the text file into a row of table cells and use Constants.FS_CellSeparator (29) and Constants.FS_NumCellSeparators (30) to determine how to divide the line into separate cells, if Constants.FS_FileIsText (16) is set to Constants.FV_DoImportAsTable (8). Constants.FS_TreatParaAsRow (28) is set to either True or False.</p> <ul style="list-style-type: none"> <li>• True - Convert each line into a row of table cells.</li> <li>• False - Convert each line into a table cell instead.</li> </ul>

When you import text into a table, in addition to setting Constants.FS\_FileIsText (16) to Constants.FV\_DoImportAsTable (8), you must specify a value for the Constants.FS\_ImportTblTag (25) property. If you set the Constants.FS\_TreatParaAsRow (28) property to True, you must also specify a value for the Constants.FS\_CellSeparator (29) property.

The property list returned by GetImportDefaultParams() does not specify values for the Constants.FS\_ImportTblTag (25) and Constants.FS\_CellSeparator (29) properties.

If you use the property list to import a table and do not specify a value for Constants.FS\_ImportTblTag (25), Import() fails and sets FA\_errno to Constants.FE\_BadParameter (-43).

If you set Constants.FS\_TreatParaAsRow (28) to True and do not specify a cell separator by setting Constants.FS\_CellSeparator (29), Import() fails and sets FA\_errno to Constants.FE\_BadParameter (-43).

If GetImportDefaultParams() fails, the Import() method sets the len field of the returned PropVals structure to 0.

The Import() method returns the object that was imported, on success.

**Note:** Import() returns 0 on success, if it imports a copy or a graphic inset.

On failure, the method assigns one of the following values to FA\_errno:

Error	Reason
Constants.FE_SystemError (-42)	System error, such as an unreadable file or insufficient memory.
Constants.FE_BadParameter (-43)	The property list contained an invalid parameter.
Constants.FE_BadFileType (-83)	The specified file exists, but it does not have the correct file type.
Constants.FE_MissingFile (-84)	The specified file does not exist.
Constants.FE_NoSuchFlow (-82)	The script specifies an import flow that does not exist.
Constants.FE_FailedState (-45)	Internal error.
Constants.FE_CircularReference (-81)	Importing the specified file causes a circular reference.
Constants.FE_FileClosedByClients (TBD)	The file was closed by a client before it could be imported.

The property list returned to importReturnParamspp has the following properties:

Property	Meaning
Constants.FS_ImportedFileName (1)	A string specifying the source file's path. If you scripted Constants.FS_ShowBrowser (1), this path can be different from the one you specified in the Import script.
Constants.FS_ImportNativeError (2)	The error condition; normally the same value as FA_errno. If the file is imported successfully, Constants.FS_ImportNativeError (2) is set to Constants.FE_Success (0).
Constants.FS_ImportStatus (3)	A bit field indicating what happened when the file was imported. See the following table for more information.

Both the `Constants.FS_ImportNativeError` (2) property and the `FA_errno` global variable indicate the result of a call to the `Import()` method. The following table lists the possible status flags and the reasons associated with them:

Status Flag	Reason
<ul style="list-style-type: none"> <li>• <code>Constants.FE_BadParameter</code> (-43)</li> <li>• <code>Constants.FE_BadFileType</code> (-83)</li> <li>• <code>Constants.FE_MissingFile</code> (-84)</li> <li>• <code>Constants.FE_FailedState</code> (-45)</li> <li>• <code>Constants.FE_CanceledByClient</code> (-86) (file was not imported)</li> </ul>	<ul style="list-style-type: none"> <li>• <code>Constants.FV_BadImportFileName</code> (16) - The specified source filename is invalid.</li> <li>• <code>Constants.FV_BadImportFileType</code> (17) - The Import script specified a file type different from the source file's actual type.</li> <li>• <code>Constants.FV_BadImportScriptValue</code> (18) - The Import script contained an invalid property value.</li> <li>• <code>Constants.FV_BadTextFileTypeHint</code> (29) - The file was a text file, and the string in <code>Constants.FS_FileTypeHint</code> (34) was not a valid import hint string.</li> <li>• <code>Constants.FV_MissingScript</code>: <code>Import()</code> was called without a script.</li> <li>• <code>Constants.FV_DisallowedImportType</code> (21) - Source file's type is disallowed by script.</li> <li>• <code>Constants.FV_NoMainFlow</code> (22) - A script was specified to import the main flow, but the source file does not have a main flow.</li> <li>• <code>Constants.FV_NoFlowWithSpecifiedName</code> (23) - A script was specified with a flow name that does not exist.</li> <li>• <code>Constants.FV_InsertionPointNotInText</code> (24) - The insertion point in the enclosing document is not in the text.</li> <li>• <code>Constants.FV_InsufficientMemory</code> (27) - There is insufficient memory to import the source file.</li> <li>• <code>Constants.FV_BadEnclosingDocId</code> (28) - There is no open document with the specified ID.</li> <li>• <code>Constants.FV_ImportFileNotReadable</code> (48) - The specified source file is unreadable.</li> </ul>
<p><code>Constants.FE_Success</code> (0)</p>	<ul style="list-style-type: none"> <li>• <code>Constants.FV_ImportedByCopy</code> (0) - The source file was imported by copy.</li> <li>• <code>Constants.FV_ImportTextTable</code> (TBD) - The source file is a text file, which was imported into a table.</li> <li>• <code>Constants.FV_ImportedMIF</code> (3) - The source file is a MIF file.</li> <li>• <code>Constants.FV_ImportedMakerDoc</code> (4) - The source file is a FASL file.</li> <li>• <code>Constants.FV_ImportedFilteredFile</code> (5) - The source file was filtered.</li> <li>• <code>Constants.FV_ImportedGraphicFile</code> (6) - The source file is a graphics file.</li> <li>• <code>Constants.FV_ImportedSgmlDoc</code> (8) - The source file is an SGML document.</li> <li>• <code>Constants.FV_ImportedXmlDoc</code> (9) - The source file is an XML document.</li> </ul>



Status Flag	Reason
Constants.FE_Canceled (-44)	<ul style="list-style-type: none"> <li>Constants.FV_CancelFileText (32) - The file is text, so the user or the Import script canceled the Import operation.</li> <li>Constants.FV_CancelFileGraphic (36) - The source file is a graphic, so the user or the Import script canceled the Import operation.</li> <li>Constants.FV_CancelFileMacEdition (TBD) - The source file is a Macintosh Edition, so the Import script canceled the Import operation</li> <li>Constants.FV_CancelFileDoc (33) - The file is a FASL file, so the user or the script canceled the Import operation.</li> <li>Constants.FV_CancelFileSgml (40) - The file is an SGML document, so the user or the script canceled the Import operation</li> <li>Constants.FV_CancelFileXml (41) - The file is an XML document, so the user or the script canceled the Import operation.</li> <li>Constants.FV_CancelFileMIF (34) - The source file is a MIF file, so the user or the script canceled the Import operation.</li> <li>Constants.FV_CancelFileFilterable (35) - The source file is a filterable file, so the user or the script canceled the Import operation.</li> <li>Constants.FV_InsertionPointInFootnote (26) - The insertion point was in a footnote and the import script specified to import the file as a table, so the file could not be imported.</li> <li>Constants.FV_InsertionPointInTableCell (25) - The insertion point was in a table cell and the import script specified to import the file as a table, so the file could not be imported</li> <li>Constants.FV_UserCanceledImport (37) - The user canceled the Import operation.</li> <li>Constants.FV_UserCanceledImportBrowser (TBD): The user canceled the Import browser</li> </ul>

To determine whether a particular `Constants.FS_ImportStatus` (3) bit is set, use `CheckStatus()`.

**Returns**

Object

**Syntax**

`Import(textLocP, filename, importParams, importReturnParams)`

**Parameters**

Parameter name	Data Type	Optional	Description
textLocP	<a href="#">TextLoc</a>	No	The text location at which to import the file.
filename	string	No	The full path of the file to import.
importParams	<a href="#">PropVals</a>	No	A property list specifying how to import the file and how to respond to errors and other conditions. To use the default list, specify null.
importReturnParams	<a href="#">PropVals</a>	No	A property list that provides information about how FrameMaker imported the file. This parameter must be initialized before you call <code>Import()</code> .

## GetProps

### Description

Retrieves the properties of the specified Book.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the value of the specified property of the Book.

See “[SetProps](#)” on page 454 under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The value to set.

## ObjectValid

### Description

Indicates whether the specified book is valid. Returns 1 if the specified book is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

# BookComponent

## Delete

### Description

Deletes the specified BookComponent object.

See “Delete” on page 453 under the AFrame class for more information.

### Returns

int

### Syntax

Delete()

## MoveComponent

### Description

Moves a particular book component up and down in the book components sequence. In addition , you can promote and demote operations of a component in a hierarchy using this method.

### Returns

Void

### Syntax

MoveComponent (moveAction)

### Parameters

Parameter name	Data Type	Optional	Description
moveAction	int	No	TBD

## GetProps

### Description

Retrieves the properties of the specified BookComponent object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

GetProps()

## SetProps

### Description

Sets the property of the BookComponent object.

See “[SetProps](#)” on page 454 under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The value of the property to be set.

## ObjectValid

### Description

Indicates whether a BookComponent object is valid. Returns 1 if the specified book component is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Cell

### StraddleCells

#### Description

The `StraddleCells()` method straddles the specified cells in a table.

The cells you straddle must all be from the same type of row. You cannot straddle a set of cells that are in both heading and body rows or footing and body rows. Also, the cells you straddle must be unstraddled. You cannot use this function to further straddle cells that are already straddled.

The `StraddleCells()` method returns `FE_Success` on success.

The method assigns one of the following values to `FA_errno`, on failure:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker product does not support tables.
Constants.FE_BadOperation (-27)	Parameters specify an action that is invalid.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.
Constants.FE_BadObjId (-3)	Invalid cell.

### Returns

int

### Syntax

StraddleCells (heightInRows, widthInCols)

### Parameters

Parameter name	Data Type	Optional	Description
heightInRows	int	No	Denotes the number of cells to straddle vertically.
widthInCols	int	No	Indicates the number of cells to straddle horizontally.

## UnStraddleCells

### Description

The UnStraddleCells() method unstraddles the specified cells in a table.

The UnStraddleCells() method returns `FE_Success` on success. The method assigns one of the following values to `FA_errno`, on failure:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker product does not support tables.
Constants.FE_BadOperation (-27)	Parameters specify an action that is invalid.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.
Constants.FE_BadObjId (-3)	Invalid cell.

### Returns

int

### Syntax

UnStraddleCells (heightInRows, widthInCols)

**Parameters**

Parameter name	Data Type	Optional	Description
heightInRows	int	No	The number of cells to unstraddle vertically.
widthInCols	int	No	The number of cells to unstraddle horizontally.

**GetText**

**Description**

The `GetText ()` method gets the text from the following types of objects:

- Cell
- Element
- Flow
- Fn
- Pgf
- SubCol
- TextFrame
- TextLine
- TiApiClient
- TiFlow
- TiText
- TiTextTable
- XRef

Call this method on the document as follows:

```
document.GetText (flags)
```

`GetText ()` returns an object containing an array of text items. Each text item contains either a string of text, an object that appears within the text (such as a table or an anchored frame), an indicator that the text properties have changed, or the object that organizes the text (such as a paragraph or a text column).

**Note:** Use the `DeallocateTextItems ()` method to deallocate the memory used by the `TextItems` structure returned by `GetText ()` when you have finished using it.

To get specific types of text items, OR the constants that represent them (for example, use a bitwise OR to combine `FTI_FlowBegin` and `FTI_String`) into flags. To get all types of text items, specify -1.

The method returns a `TextItems` object containing the array of text items of the requested text types.

`TextItem.dataType` can be one of the following constants:

Text Item Type	Text Item Represents	Text Item Data
<code>FTI_CharPropsChange</code>	A change in the text properties	Flags indicating which properties have changed (see the following table).
<code>FTI_ElementBegin</code>	The beginning of a container structural element	<code>FO_Element</code>

**Function Summary**

<b>Text Item Type</b>	<b>Text Item Represents</b>	<b>Text Item Data</b>
FTI_ElementEnd	The end of a container structural element	FO_Element
FTI_ElemPrefixBegin	The beginning of an element's prefix	FO_Element
FTI_ElemPrefixEnd	The end of an element's prefix	FO_Element
FTI_ElemSuffixBegin	The beginning of an element's suffix	FO_Element
FTI_ElemSuffixEnd	The end of an element's suffix	FO_Element
FTI_FlowBegin	The beginning of a flow	FO_Flow
FTI_FlowEnd	The end of a flow	FO_Flow
FTI_FnAnchor	A footnote	FO_Fn
FTI_FrameAnchor	An anchored frame	FO_AFrame
FTI_LineBegin	The beginning of a line	-
FTI_LineEnd	The end of a line and the line end type	If the line end is a normal line end, the value is 0; if it is a forced line end, the FTI_HardLineEnd flag is set; if it is a hyphen line end, the FTI_HyphenLineEnd flag is set
FTI_MarkerAnchor	A Marker	FO_Marker
FTI_PageBegin	The beginning of a page	FO_Page
FTI_PageEnd	The end of a page	FO_Page
FTI_PgfBegin	The beginning of a paragraph	FO_Pgf
FTI_PgfEnd	The end of a paragraph	FO_Pgf
FTI_String	A string of characters with the same condition and character format.	A character string
FTI_SubColBegin	The beginning of a column	FO_SubCol
FTI_SubColEnd	The end of a column	FO_SubCol
FTI_TblAnchor	A Table	FO_Tbl
FTI_TextFrameBegin	The beginning of a text frame	FO_TextFrame
FTI_TextFrameEnd	The end of a text frame	FO_TextFrame
FTI_TextInsetBegin	The beginning of a text inset	FO_TiApiClient, FO_TiFlow, FO_TiText, or FO_TiTextTable
FTI_TextInsetEnd	The end of a text inset	FO_TiApiClient, FO_TiFlow, FO_TiText, or FO_TiTextTable
FTI_TextObjId	The object to which the offsets of all the text items are relative	FO_Pgf or FO_TextLine
FTI_VarBegin	The beginning of a variable	FO_Var
FTI_VarEnd	The end of a variable	FO_Var
FTI_XRefBegin	The beginning of a cross-reference	FO_XRef
FTI_XRefEnd	The end of a cross-reference	FO_XRef

The following table lists the bit flags that a client can bitwise AND with the `idata` field of a `Constants.FTI_CharPropsChange` (0x00200000) text item. For example, to determine if the font family changed, bitwise AND the `Constants.FTF_FAMILY` (0x80000000) flag with the `idata` field.

**Function Summary**

<b>Flags</b>	<b>Meaning</b>
FTF_ALL	OR of all the flags
FTF_ANGLE	The font angle has changed.
FTF_CAPITALIZATION	The capitalization has changed.
FTF_CHANGEBAR	The change bars have changed.
FTF_CHARTAG	The Character Catalog format has changed.
FTF_COLOR	The color has changed.
FTF_CONDITIONTAG	The condition tag has changed.
FTF_ENCODING	The text encoding has changed,
FTF_FAMILY	The font family has changed.
FTF_IIF	An internal flag having to do with Asian text. input. If there is a non-zero value for this flag, a front-end processor is controlling that text and you should not modify the associated text item.
FTF_KERNX	The kern-x characteristic has changed.
FTF_KERNY	The kern-x characteristic has changed.
FTF_LANGUAGE	Character language has changed.
FTF_OUTLINE	The outline characteristic has changed.
FTF_OVERLINE	The overline characteristic has changed.
FTF_PAIRKERN	The pair kerning has changed.
FTF_POSITION	The character position has changed.
FTF_SHADOW	The shadow characteristic has changed.
FTF_SIZE	The font size has changed.
FTF_SPREAD	The font spread has changed.
FTF_STRETCH	The font stretch value has changed.
FTF_STRIKETHROUGH	The strikethrough characteristic has changed.
FTF_TSUME	The Tsume setting has changed.
FTF_UNDERLINING	The underlining has changed.
FTF_VARIATION	The font variation has changed.
FTF_WEIGHT	The font weight has changed.

On success, the method returns a `TextItems` structure containing the array of text items of the requested types.

On failure, the method sets the `len` field of the returned `TextItems` structure to 0, and returns one of the following values to `FA_errno`:

<b>Error</b>	<b>Reason</b>
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_NotTextObject (-41)	Object specified for the text range is not an object that contains text.



If you call `GetText ()` for an object (FO\_Element object), the returned information depends on the type of element, as shown:

Value	Information Returned
Constants.FV_FO_CONTAINER (1)	All the text items from the beginning to the end of the element.
Constants.FV_FO_SYS_VAR (2)	All the text items from the beginning to the end of the variable.
Constants.FV_FO_XREF (3)	All the text items from the beginning to the end of the cross reference.
Constants.FV_FO_FOOTNOTE (5)	All the text items from the beginning to the end of the footnote.
Constants.FV_FO_TBL_TITLE (9)	All the text items from the beginning to the end of the table title.
Constants.FV_FO_TBL_CELL (14)	All the text items from the beginning to the end of the cell.
Constants.FV_FO_TBL_HEADING (10)	Nothing. GetText() failed.
Constants.FV_FO_TBL_BODY (11)	
Constants.FV_FO_TBL_FOOTING (12)	
Constants.FV_FO_MARKER (4)	
Constants.FV_FO_TBL (8)	
Constants.FV_FO_GRAPHIC (6)	
Constants.FV_FO_EQN (7)	
Constants.FV_FO_TBL_ROW (13)	

**Returns**

[TextItems](#)

**Syntax**

`GetText (flags)`

**Parameters**

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

**GetProps**

**Description**

Retrieves the properties of the specified cell object.

See “[GetProps](#)” on page 453 under the AFrame class for more information.

**Returns**

[PropVals](#)

**Syntax**

`GetProps ()`

## SetProps

### Description

Sets the desired property of the cell object.

See “[SetProps](#)” on page 454 under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The value of the property to set.

## ObjectValid

### Description

Indicates whether a cell object is valid. Returns 1 if the cell object is valid, else returns 0.

See “[ObjectValid](#)” on page 455 under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## CharFmt

### Delete

#### Description

Deletes the specified CharFmt object.

See “[Delete](#)” on page 453 under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

## GetProps

### Description

Get the properties of the specified CharFmt object.

See “[GetProps](#)” on page 453 under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the value of the property for the specified CharFmt object.

See “[SetProps](#)” on page 454 under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The value of the property to set.

## ObjectValid

### Description

Indicates whether the specified CharFmt object is valid. Returns 1 if the CharFmt object is valid, else returns 0.

See “[ObjectValid](#)” on page 455 under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

# CMSObject

## CMSsetProperty()

### Description

Sets a single property for a CMS object.

### Returns

void

### Syntax

```
void CMSsetProperty (cmsSessionId,objectId,propval);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
propVal	int	No	The specified property of a CMS object as a PropVal data structure

If CMSsetProperty() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_CMSRootObjectExists	The method tries to set a root which already exists.
Constants.FE_CMSBadItemType	The file type expected by the cms object does not match the valid file type.
Constants.FE_CMSBadItemContainerType	The container value expected by the cms object is not properly set

## CMSsetPropertyProperties()

### Description

Sets multiple properties of a CMS object

### Returns

void

### Syntax

```
CMSsetPropertyProperties (cmsSessionId,objectId,propVals);
```

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
propVals	int	No	A property list that specifies the CMS properties . Properties are added as propvals with the identifier as CMSItemPropertyT enum and value.

If CMSSetProperties() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_CMSRootObjectExists	The method tries to set a root which already exists.
Constants.FE_CMSBadItemType	The file type expected by the cms object does not match the valid file type.
Constants.FE_CMSBadItemContainerType	The container value expected by the CMS object is not properly set

**CMSGetProperty()**

**Description**

Gets a specified property of a CMS object.

**Returns**

[PropVals](#)

**Syntax**

`CMSGetProperty (cmsSessionId, objectId,propertyId) ;`

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session.
objectId	int	No	The ID of the CMS Object.
propertyId	int	No	F_PropldentT which allows user to specify property identifier as Integer value OR string value based on the CMS. For e.g. Documentum works on object name, whereas Generic CMS works on Integer based identifier ID of the CMS Object.

If CMSGetProperty() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.

Error	Reason
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSGetProperties()

### Description

Gets the properties of a CMS object.

### Returns

[PropVals](#)

### Syntax

```
CMSGetProperties (cmsSessionId, objectId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
objectId	int	No	The ID of the CMS Object

If CMSGetProperties() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSErrorBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSErrorBadObjectId	The client specified an invalid cms object ID.

## CMSEnableCommand()

### Description

Enables the specified CMS command in the context menu of the CMS tree within FrameMaker.

### Returns

void

### Syntax

```
CMSEnableCommand (cmsSessionId, objectId, commandId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
commandId	int	No	The command to enable

If CMSEnableCommand() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_CMSBadCommandId	The client specified an invalid command ID.

## CMSTDisableCommand()

### Description

Disables the specified CMS command in the context menu of the CMS tree within FrameMaker.

### Returns

void

### Syntax

```
CMSTDisableCommand (cmsSessionId, objectId, commandId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
commandId	int	No	The command to disable

If CMSTDisableCommand() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_CMSBadCommandId	The client specified an invalid command ID.

## CMSShowCheckoutUI()

### Description

Displays the checkout dialog for a CMS object

### Returns

bool

### Syntax

```
CMSShowCheckoutUI (sessionId, objectId, hideUiItems);
```

**Parameters**

Parameter name	Data Type	Optional	Description
sessionId	int	No	The ID of the CMS session
objectId	int	No	The ID of the CMS Object
hideUiItems	int	No	Parameter to customize the checkout dialog. It can take the values from CMSCustomizeCheckoutUI enum

Enum constants used to customize CMS Object's Checkout user interface.

The possible values of the hideUiItems field are

Value for flags	Meaning
FV_CMSCheckoutUI_Id_ShowDependents	Flag to hide "Show dependents" checkbox

If CMSShowCheckoutUI() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID
Constants.FE_BadParameter	The function call specified an invalid parameter.

**CMSShowCheckinUI()**

**Description**

Displays the checkin dialog for a CMS object

**Returns**

bool

**Syntax**

CMSShowCheckinUI (sessionId, objectId, hideUiItems);

**Parameters**

Parameter name	Data Type	Optional	Description
sessionId	int	No	The ID of the CMS session
objectId	int	No	The ID of the CMS Object
hideUiItems	int	No	Parameter to customize the checkout dialog. It can take the values from CMSCustomizeCheckinUI enum

Enum constants used to customize CMS Object's Checkin user interface.

The possible values of the hideUiItems field are



hideUIItems constants	Reason
Constants.FV_CMSCheckinUI_Id_MajorVersion	Flag to hide "Major version" radio button
Constants.FV_CMSCheckinUI_Id_VersionLabel	Flag to hide "Version Label" text field
Constants.FV_CMSCheckinUI_Id_Description	Flag to hide "Description" text field
Constants.FV_CMSCheckinUI_Id_CheckinComment	Flag to hide "Checkin comment" text field
Constants.FV_CMSCheckinUI_Id_MakeThisCurrentVersion	Flag to hide "Make this current version" checkbox

If CMSShowCheckinUI() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSShowCancelCheckoutUI()

### Description

Displays the Cancel Check out dialog for a CMS object

### Returns

void

### Syntax

```
CMSShowCancelCheckoutUI (sessionId, objectId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session
cmsObjectId	int	No	The ID of the CMS object

If CMSShowCancelCheckoutUI() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID

## CMSShowDeleteUI()

### Description

Displays the Delete dialog for a CMS object

### Returns

CMSShowDeleteParam

### Syntax

```
CMSShowDeleteUI (F_ObjHandleT cmsSessionId, objectId, T hideUiItems);
```

### Parameters

Parameter name	Data Type	Optional	Description
sessionId	int	No	The ID of the CMS session
objectId	int	No	The ID of the CMS Object
hideUiItems	int	No	Parameter to customize the delete dialog. It can take the values from CMSCustomizeDeleteUI enum

Enum constants used to customize CMS Object's Delete user interface.

The user can specify one or more of the following flag constants (using the OR expression for multiple flags) into the hideUiItems field:

Value for flags	Meaning
Constants.FV_CMSDeleteUI_DeleteAllVersion	Flag to hide "Delete all the version of a file" checkbox
Constants.FV_CMSDeleteUI_DeleteAllDependents	Flag to hide "Delete all the dependents of a file" checkbox

If CMSShowDeleteUI() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSShowCommonListUI()

### Description

Displays the list-based dialogs such as "Show Version", "Show Checked out files", "Show dependents" and "Show result". These items are fetched using the CMSGetItems CMS command.

### Returns

bool

**Syntax**

`CMSShowCommonListUI (cmsSessionId, objectId, commandId, title, columnProperties);`

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
commandId	int	No	The ID of the CMS command
title	int	No	The Title of the Show Dialog
columnProperties	int	No	Properties Columns to show

If `CMSShowCommonListUI()` fails, the method assigns following values to `FA_errno`:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_CMSBadCommandId	The client specified an invalid command ID.
Constants.FE_BadParameter	The function call specified an invalid parameter.

**CMSShowPropertyUI()**

**Description**

Displays the Property dialog for a CMS object

**Returns**

`CMSProperties`

**Syntax**

`CMSShowPropertyUI ( cmsSessionId, objectId, props);`

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
props	int	No	CMSPropertiesT structure specifies a set of CMS object properties. Set NULL if user wants to show default properties

If `CMSShowPropertyUI()` fails, the method assigns following values to `FA_errno`:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID

Error	Reason
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSShowPropertyUIWithTitle

### Description

Displays the Property dialog for a CMS object with a specified title.

### Returns

CMSProperties

### Syntax

```
CMSShowPropertyUIWithTitle ( cmsSessionId, objectId, props, title);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
props	int	No	CMSPropertiesT structure specifies a set of CMS object properties. Set NULL if user wants to show default properties
title	string	No	Title of the dialog box

If CMSShowPropertyUIWithTitle fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSProperties

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to concat(), each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

### Returns

[Ints](#)

### Syntax

concat (value)

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Ints</a>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

int

### Syntax

pop ()

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

push (value)

### Parameters

Parameter name	Data Type	Optional	Description
value	int	No	TBD

## CMSRegistration

### CMSGetCMSInfo()

### Description

Gets the CMS information for a particular CMS registration id.

### Returns

CMSInfo

### Syntax

```
CMSGetCMSInfo (cmsId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsId	int	No	The registration ID of the CMS

If CMSGetCMSInfo() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadParameter	The function call specified an invalid parameter.

## CMSConfigLoginUI()

### Description

Configures the CMS Connector Manager dialogs within the FrameMaker interface

### Returns

void

### Syntax

```
CMSConfigLoginUI (cmsId, userFields, userLoginUI);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsId	int	No	The ID of the CMS
userFields	int	No	Optional user fields with strings. User can add upto two user fields.
userLoginUI	int	No	1 if user want to impement his own custom connection manager dialog

If CMSConfigLoginUI() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadParameter	The function call specified an invalid parameter.

## GetProps

### Description

Get the properties of the specified CMSRegistration object.

See “[GetProps](#)” on page 453 under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the value of the property for the specified CMSRegistration object.

See “[SetProps](#)” on page 454 under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The value of the property to set.

## ObjectValid

### Description

Indicates whether the specified CMSRegistration object is valid. Returns 1 if the CMSRegistration object is valid, else returns 0.

See “[ObjectValid](#)” on page 455 under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## CMSSession

### CMSLogout()

### Description

Logs out the user from a particular CMS connection

**Returns**

void

**Syntax**

CMSLogout(cmsSessionId);

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session

If CMSLogout() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSFailedLogout	Failed to log out of the CMS

## GetCMSObjectFromPath()

**Description**

Gets CMS object from a URL path

**Returns**

CMS Object

**Syntax**

GetCMSObjectFromPath (cmsSessionId, urlPath)

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
urlPath	int	No	The url pathname of the file or folder

If GetCMSObjectFromPath() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSFailedGetItemFromPath	Failed to get the CMS object from the specified path

## CMSShowBrowseRepositoryUI

**Description**

Displays repository browser dialog based on flag “showContainerOnly”

**Returns**

CMS Object



### Syntax

```
CMSShowBrowseRepositoryUI (cmsSessionId,showContainerOnly);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session
showContainerOnly	int	No	1 if only container item is shown 0 if all items are shown

## CMSCreateObject

### Description

Creates a CMS object.

### Returns

CMS Object

### Syntax

```
CMSCreateObject (cmsSessionId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session

If CMSCreateObject() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSSessionId	The client specified an invalid session ID.
Constants.FE_CMSCreationFailed	Method failed to create a cms object.

## CMSSessionIdFromSession

### Description

Gets the CMS registration id from CMS Session id

### Returns

CMS Object

### Syntax

```
F_ApiCMSSessionIdFromSession (cmsSessionId);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session

If CMSGetCmsIdFromSession() fails, the method assigns following values to FA\_erno:

Error	Reason
Constants.FE_CMSErrorBadSessionId	The client specified an invalid session ID

## Color

### Delete

#### Description

Deletes the specified color object.

See “Delete” on page 453 under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

### GetProps

#### Description

Retrieves the properties of the specified color object.

See “GetProps” on page 453 under the AFrame class for more information.

#### Returns

[PropVals](#)

#### Syntax

```
GetProps()
```

### SetProps

#### Description

Sets the value of the property for the color object.

See “SetProps” on page 454 under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The value of the property to be set.

## ObjectValid

### Description

Determines whether a color object is valid. Returns 1 if the Color object is valid, else returns 0.

See “[ObjectValid](#)” on page 455 under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## CombinedFont

### CombinedFont

### Description

`CombinedFont` specifies a set of font characteristics for a combined font. The `combinedFont` field specifies a `CombinedFont` object. From this object you can get information about the combined font such as the base and Western font families, or the combined font name.

### Returns

[CombinedFont](#)

### Syntax

```
CombinedFont (combinedFont, variation, weight, angle)
```

### Parameters

Parameter name	Data Type	Optional	Description
combinedFont	<a href="#">CombinedFontDefn</a>	No	CombinedFont object.
variation	uint	No	Index of the font variation.
weight	uint	No	Index of the font weight.
angle	uint	No	Index of the font angle.

## CombinedFontDefn

### CombinedFamilyFonts

#### Description

Returns the permutations of angles, variations, and weights available for a specified combined font definition.

The method does not take any arguments.

The `CombinedFamilyFonts()` method returns a `CombinedFonts` object that provides a list of the permutations of angles, variations, and weights available for the specified combined font definition.

#### Returns

[CombinedFonts](#)

#### Syntax

```
CombinedFamilyFonts()
```

## Delete

#### Description

Deletes the specified `CombinedFonts` object.

See “Delete” on page 453 under the `AFrame` class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

## GetProps

#### Description

Retrieves the properties of the specified `CombinedFonts` object.

**Function Summary**

See “[GetProps](#)” on page 453 under the AFrame class for more information.

**Returns**

[PropVals](#)

**Syntax**

```
GetProps()
```

## SetProps

**Description**

Sets the value of the desired property of the CombinedFonts object.

See “[SetProps](#)” on page 454 under the AFrame class for more information.

**Returns**

Void

**Syntax**

```
SetProps(setVal)
```

**Parameters**

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The value of the property to be set.

## ObjectValid

**Description**

Indicates whether a CombinedFonts object is valid. Returns 1 if the CombinedFonts object is valid, else returns 0.

See “[ObjectValid](#)” on page 455 under the AFrame class for more information.

**Returns**

int

**Syntax**

```
ObjectValid()
```

## CombinedFonts

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

## concat

### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

### Returns

`CombinedFonts`

### Syntax

`concat (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<code>CombinedFonts</code>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

`CombinedFont`

### Syntax

`pop ()`

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

`push (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<code>CombinedFont</code>	No	The value to push into the array

# Command

## Delete

### Description

Deletes the specified Command object.

See “[Delete](#)” on page 453 under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the specified Command object.

See “[GetProps](#)” on page 453 under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the value of the specified Command property.

See “[SetProps](#)” on page 454 under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The value of the property to set.

## ObjectValid

### Description

Identifies whether a Command object is valid or not. Returns 1 if the Command object is valid, else returns 0.

See “[ObjectValid](#)” on page 455 under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Delete

### Description

Deletes the specified Command object.

The method does not take any arguments.

Call the Delete() method directly on the object to be deleted.

### Returns

int

### Syntax

```
Delete()
```

## CompareRet

### CompareRet

#### Description

CompareRet provides the results of a call to the Compare() method. The method returns a CompareRet object with the summary and composite documents.

#### Returns

[CompareRet](#)

#### Syntax

```
CompareRet(sum, comp)
```



### Parameters

Parameter name	Data Type	Optional	Description
sum	<a href="#">Doc</a>	No	Summary Document.
comp	<a href="#">Doc</a>	No	Composite Document.

## CondFmt

### Delete

#### Description

Deletes the specified `CondFmt` object.

See “Delete” on page 453 under the `AFrame` class for more information.

#### Returns

`int`

#### Syntax

```
Delete()
```

## DeleteCondTag

#### Description

Deletes a conditional tag from a document.

#### Returns

`Status`

#### Syntax

```
DeleteCondTag(docId, condTagId, action);
```

### Parameters

Parameter name	Data Type	Optional	Description
docId	<code>int</code>	No	The id of the document containing the conditional tag to be deleted
condTagId	<code>int</code>	No	The id of the conditional tag object (type: <code>FO_CondFmt</code> ) in the doc
action	<code>int</code>	No	One of the following: <code>FF_UNTAGGED_ASK</code> : Prompt the user <code>FF_UNTAGGED_UNCOND</code> : Make text unconditional <code>FF_UNTAGGED_DELETE</code> : Delete text

If DeleteCondTag() fails, the method assigns following values to FA\_erno:

Error	Reason
Constants.FE_ReadOnly	Document is read only.
Constants.FE_BadValue	Action is not one of the specified values
Constants.FE_Success	Deletion was successful

## GetProps

### Description

Retrieves the properties of the specified CondFmt object.

See “[GetProps](#)” on page 453 under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

GetProps ()

## SetProps

### Description

Sets the value of the specified CondFmt object property.

See “[SetProps](#)” on page 454 under the AFrame class for more information.

### Returns

Void

### Syntax

SetProps (setVal)

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The value of the property to set.

## ObjectValid

### Description

Identifies whether a CondFmt object is valid or not. Returns 1 if the CondFmt object is valid, else returns 0.

See “[ObjectValid](#)” on page 455 under the AFrame class for more information.

### Returns

int

### Syntax

ObjectValid()

## Doc

### AddText

#### Description

Inserts text into a paragraph or a text line.

The text you specify for text must use the FrameMaker character set. To add special characters, you must specify octal (\) or hexadecimal (\x) sequences. The following table lists some of these sequences.

Special Character	Hexadecimal Representation	Octal Representation
>	\x3e	\76
" (straight double quotation mark)	\x22	\42
" (left double quotation mark)	\xd2	\322
" (right double quotation mark)	\xd3	323

The method returns the location of the text that was added, on success. On failure, it sets one of the following error numbers to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object.
Constants.FE_NotTextObject (-41)	The object that textLoc specifies is not a paragraph (FO_Pgf) or a text line (FO_TextLine).
Constants.FE_OffsetNotFound (-21)	The offset specified for the text location could not be found in the specified text object.
Constants.FE_ReadOnly (-6)	The document is read only.
Constants.FE_BadSelectionForOperation (-59)	The location that textLoc specifies is invalid. For example, it is inside a variable or outside the highest level element in a structured FrameMaker document.

### Returns

[TextLoc](#)

### Syntax

AddText (textLoc, text)

### Parameters

Parameter name	Data Type	Optional	Description
textLoc	<a href="#">TextLoc</a>	No	The text location at which to add the text.
text	string	No	The text to add.

## CenterOnText

### Description

Centers a range of text such that the middle of the text appears in the middle of the document window.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadRange (-36)	The specified text range is invalid.
Constants.FE_NotTextObject (-41)	The object that <code>textRange</code> specifies is not a paragraph ( <code>FO_Pgf</code> ) or a text line ( <code>FO_TextLine</code> ).
Constants.FE_OffsetNotFound (-21)	The offset specified for the text location could not be found in the specified paragraph or text line.

### Returns

int

### Syntax

`CenterOnText (textRange)`

### Parameters

Parameter name	Data Type	Optional	Description
textRange	<a href="#">TextRange</a>	No	The range of text to center.

## Clear

### Description

Deletes the current selection from a document.

If you specify 0 for flags, the method suppresses any interactive alerts or warnings that arise, leaves the selected table cells empty, and deletes hidden text.

You can OR the following values into flags:

Flags	Meaning
Constants.FF_INTERACTIVE (0x0001)	Prompt the user with dialog or alert boxes that arise.
Constants.FF_CUT_TBL_CELLS (0x0002)	Remove cleared table cells.
Constants.FF_VISIBLE_ONLY (0x0010)	Clear only the visible portion of the selection.

**Function Summary**

Flags	Meaning
Constants.FF_DONT_DELETE_HIDDEN_TEXT (0x0004)	Do not delete hidden text.

The `Constants.FF_INTERACTIVE` flag takes precedence over other flags. If you specify `Constants.FF_INTERACTIVE` | `Constants.FF_DONT_DELETE_HIDDEN_TEXT` and the selection contains hidden text, then FrameMaker allows you to select whether to delete the hidden text or to retain it.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current selection is invalid for this operation.
Constants.FE_Canceled (-44)	The User or parameters canceled the operation.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation.

**Returns**

int

**Syntax**

`Clear(flags)`

**Parameters**

Parameter name	Data Type	Optional	Description
flags	int	No	Bit field that specifies how to clear the text and how to handle interactive alerts. For default settings, specify 0.

## ClearAllChangebars

**Description**

Clears all change bars from a specified document.

**Note:** The method performs the same function as clicking the *Clear All Change Bars* box in the *Change Bars* dialog box.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation.
Constants.FE_SystemError (-42)	A system error occurred.

**Returns**

int

**Syntax**

`ClearAllChangebars()`

## Compare

### Description

Compares two documents or two books.

You can OR the values shown in the following table into the flags argument.

Flags	Meaning
Constants.FF_CMP_SUMMARY_ONLY (0x01)	Generate a summary document, but not a composite document.
Constants.FF_CMP_CHANGE_BARS (0x02)	Turn on change bars in the composite document.
Constants.FF_CMP_HYPERLINKS (0x04)	Put hypertext links in the summary document.
Constants.FF_CMP_SUMKIT (0x08)	Open the summary document.
Constants.FF_CMP_COMPKIT (0x10)	Open the composite document.

On success, the method returns a `CompareRet` structure containing the summary and composite documents detailing the changes found, if any.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadCompare (-35)	Documents to be compared are not of the same type.
Constants.FE_CompareTypes (-34)	One of the files is not a FrameMaker document or book, or one file is a book and the other is a document.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation.

### Returns

[CompareRet](#)

### Syntax

`Compare(newerDoc, flags, insertCondTag, deleteCondTag, replaceText, compareThreshold)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>newerDoc</code>	<code>Doc</code>	No	The newer version of the document to be compared.
<code>flags</code>	<code>int</code>	No	Bit flags that specify how to generate the summary and composite documents. Specify 0 for the default flags.
<code>insertCondTag</code>	<code>string</code>	No	The condition tag to apply to insertions shown in the composite document. For no insert condition tag, specify <code>null</code> .

Parameter name	Data Type	Optional	Description
deleteCondTag	string	No	The condition tag to apply to deletions shown in the composite document. For no delete condition tag, specify <code>null</code> .
replaceText	string	No	Text to appear in place of the deleted text. For no replacement text, specify <code>null</code> .
compareThreshold	int	No	Threshold expresses the percentage of words that can change before paragraphs are considered not equal. If two paragraphs are equal, word differences between them are shown within a paragraph in the composite document.  If this value is set to zero, the compare function uses the default value of 75. If not zero, the function uses the setting in the FrameMaker configuration files.

## Copy

### Description

Copies the current selection to the FrameMaker Clipboard.

If the value of the flags parameter is 0, the method suppresses any interactive alerts or warnings that arise.

You can also OR the following values into flags:

Flags	Meaning
Constants.FF_INTERACTIVE (0x0001)	Prompt user with dialog or alert boxes that arise.
Constants.FF_STRIP_HYPERTEXT (0x0080)	Do not copy any hypertext markers in the selection.
Constants.FF_VISIBLE_ONLY (0x0010)	Copy only the visible portion of the selection.

The `Constants.FF_INTERACTIVE (0x0001)` flag takes precedence over other flags. If you specify `Constants.FF_INTERACTIVE (0x0001) | Constants.FF_VISIBLE_ONLY (0x0010)` and the selection is not visible, then FrameMaker allows the user to choose whether to copy the selection or not.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation
Constants.FE_BadSelectionForOperation (-59)	The Selection doesn't support the requested operation.
Constants.FE_Canceled (-44)	The User or parameters canceled the operation.
Constants.FE_BadOperation (-27)	The parameters specified an invalid operation.

### Returns

int

### Syntax

`Copy (flags)`

**Parameters**

Parameter name	Data Type	Optional	Description
flags	int	No	Bit field that specifies how to copy the text and how to handle interactive alerts. For default settings, specify 0.

**Cut**

**Description**

Cuts the current selection to the FrameMaker Clipboard.

If the value of the flags parameter is 0, the method suppresses any interactive alerts or warnings that arise.

You can OR the following values into flags:

Flags	Meaning
Constants.FF_INTERACTIVE (0x0001)	Prompt user with dialog or alert boxes that arise.
Constants.FF_CUT_TBL_CELLS (0x0002)	Remove cut table cells.
Constants.FF_VISIBLE_ONLY (0x0010)	Cut only the visible portion of the selection.
Constants.FF_DONT_DELETE_HIDDEN_TEXT (0x0004)	Do not cut hidden text.

The `Constants.FF_INTERACTIVE (0x0001)` flag takes precedence over other flags. If you specify `Constants.FF_INTERACTIVE (0x0001) | Constants.FF_DONT_DELETE_HIDDEN_TEXT (0x0004)` and the selection contains hidden text, then FrameMaker allows the user to choose whether to delete the hidden text or retain it.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation
Constants.FE_BadSelectionForOperation (-59)	The Selection doesn't support the requested operation.
Constants.FE_Canceled (-44)	The User or parameters canceled the operation.
Constants.FE_BadOperation (-27)	The parameters specified an invalid operation.

**Returns**

int

**Syntax**

Cut (flags)

**Parameters**

Parameter name	Data Type	Optional	Description
flags	int	No	Bit field that specifies how to cut the text and how to handle interactive alerts. For default settings, specify 0.



## DeleteText

### Description

Deletes a specified text range from a document.

The method returns `FE_Success` if it succeeds. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDelete (-17)	Specified text could not be deleted.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_NotTextObject (-41)	Object specified for the text range is not an object that contains text, for example, a text frame (FO_TextFrame), a paragraph (FO_Pgf) or a text line (FO_TextLine).
Constants.FE_BadSelectionForOperation (-59)	Selection is within a locked text range.

### Returns

int

### Syntax

DeleteText (textRange)

### Parameters

Parameter name	Data Type	Optional	Description
textRange	<a href="#">TextRange</a>	No	The text range to delete.

## DeleteUndefinedAttribute

### Syntax

## DemoteElement

### Description

Demotes the selected structural element or elements. The element becomes a child of the sibling element before it.

**Note:** *At least one structural element must be selected in the document, before calling DemoteElement().*

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document.

**Function Summary**

Error	Reason
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.

**Returns**

Void

**Syntax**

DemoteElement ()

**GetTextForRange****Description**

Gets the text for a specified text range.

Call the method on the document as follows:

document.GetTextForRange ()

You can OR the following constants into the flags parameter, to retrieve various text items:

Text Item Type	Text Item Represents	Text Item Data
Constants.FTI_CharPropsChange (0x00200000)	A change in the text properties	Flags indicating which properties have changed (see the following table).
Constants.FTI_ElementBegin (0x00080000)	The beginning of a container structural element	FO_Element
Constants.FTI_ElementEnd (0x00100000)	The end of a container structural element	FO_Element
Constants.FTI_ElemPrefixBegin (0x08000000)	The beginning of an element's prefix	FO_Element
Constants.FTI_ElemPrefixEnd (0x10000000)	The end of an element's prefix	FO_Element
Constants.FTI_ElemSuffixBegin (0x20000000)	The beginning of an element's suffix	FO_Element
Constants.FTI_ElemSuffixEnd (0x40000000)	The end of an element's suffix	FO_Element
Constants.FTI_FlowBegin (0x00000020)	The beginning of a flow	FO_Flow
Constants.FTI_FlowEnd (0x00000040)	The end of a flow	FO_Flow
Constants.FTI_FnAnchor (0x00001000)	A footnote	FO_Fn
Constants.FTI_FrameAnchor (0x00000800)	An anchored frame	FO_AFrame
Constants.FTI_LineBegin (0x00000002)	The beginning of a line	-

**Function Summary**

<b>Text Item Type</b>	<b>Text Item Represents</b>	<b>Text Item Data</b>
Constants.FTI_LineEnd (0x00000004)	The end of a line and the line end type	If the line end is a normal line end, the value is 0; if it is a forced line end, the FTI_HardLineEnd flag is set; if it is a hyphen line end, the FTI_HyphenLineEnd flag is set
Constants.FTI_MarkerAnchor (0x00004000)	A Marker	FO_Marker
Constants.FTI_PageBegin (0x00000080)	The beginning of a page	FO_Page
Constants.FTI_PageEnd (0x00000100)	The end of a page	FO_Page
Constants.FTI_PgfBegin (0x00000008)	The beginning of a paragraph	FO_Pgf
Constants.FTI_PgfEnd (0x00000010)	The end of a paragraph	FO_Pgf
Constants.FTI_String (0x00000001)	A string of characters with the same condition and character format.	A character string
Constants.FTI_SubColBegin (0x00000200)	The beginning of a column	FO_SubCol
Constants.FTI_SubColEnd (0x00000400)	The end of a column	FO_SubCol
Constants.FTI_TblAnchor (0x00002000)	A Table	FO_Tbl
Constants.FTI_TextFrameBegin (0x00400000)	The beginning of a text frame	FO_TextFrame
Constants.FTI_TextFrameEnd (0x00800000)	The end of a text frame	FO_TextFrame
Constants.FTI_TextInsetBegin (0x02000000)	The beginning of a text inset	FO_TiApiClient, FO_TiFlow, FO_TiText, or FO_TiTextTable
Constants.FTI_TextInsetEnd (0x04000000)	The end of a text inset	FO_TiApiClient, FO_TiFlow, FO_TiText, or FO_TiTextTable
Constants.FTI_TextObjId (0x01000000)	The object to which the offsets of all the text items are relative	FO_Pgf or FO_TextLine
Constants.FTI_VarBegin (0x00020000)	The beginning of a variable	FO_Var
Constants.FTI_VarEnd (0x00040000)	The end of a variable	FO_Var
Constants.FTI_XRefBegin (0x00008000)	The beginning of a cross-reference	FO_XRef
Constants.FTI_XRefEnd (0x00010000)	The end of a cross-reference	FO_XRef

The following table lists the flags for `FTI_CharPropsChange`. These flags indicate which properties have changed.

<b>Flags</b>	<b>Meaning</b>
Constants.FTF_ALL (0xFFFFFE03)	OR of all the flags

**Function Summary**

<b>Flags</b>	<b>Meaning</b>
Constants.FTF_ANGLE (0x10000000)	The font angle has changed.
Constants.FTF_CAPITALIZATION (0x000040000)	The capitalization has changed.
Constants.FTF_CHANGEBAR (0x01000000)	The change bars have changed.
Constants.FTF_CHARTAG (0x00008000)	The Character Catalog format has changed.
Constants.FTF_COLOR (0x00010000)	The color has changed.
Constants.FTF_CONDITIONTAG (0x00000001)	The condition tag has changed.
Constants.FTF_ENCODING (0x00000200)	The text encoding has changed,
Constants.FTF_FAMILY (0x80000000)	The font family has changed.
Constants.FTF_IIF (0x00000002)	An internal flag having to do with Asian text. input. If there is a non-zero value for this flag, a front-end processor is controlling that text and you should not modify the associated text item.
Constants.FTF_KERNX (0x00080000)	The kern-x characteristic has changed.
Constants.FTF_KERNY (0x00040000)	The kern-x characteristic has changed.
Constants.FTF_LANGUAGE (0x00000800)	Character language has changed.
Constants.FTF_OUTLINE (0x00800000)	The outline characteristic has changed.
Constants.FTF_OVERLINE (0x02000000)	The overline characteristic has changed.
Constants.FTF_PAIRKERN (0x00200000)	The pair kerning has changed.
Constants.FTF_POSITION (0x00002000)	The character position has changed.
Constants.FTF_SHADOW (0x00400000)	The shadow characteristic has changed.
Constants.FTF_SIZE (0x00100000)	The font size has changed.
Constants.FTF_SPREAD (0x00020000)	The font spread has changed.
Constants.FTF_STRETCH (0x00001000)	The font stretch value has changed.
Constants.FTF_STRIKETHROUGH (0x04000000)	The strikethrough characteristic has changed.
Constants.FTF_TSUME (0x00000400)	The Tsume setting has changed.

Flags	Meaning
Constants.FTF_UNDERLINING (0x08000000)	The underlining has changed.
Constants.FTF_VARIATION (0x40000000)	The font variation has changed.
Constants.FTF_WEIGHT (0x20000000)	The font weight has changed.

On success, the method returns a `TextItems` structure containing the array of text items of the requested types.

On failure, the method sets the `len` field of the returned `TextItems` structure to 0, and returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_NotTextObject (-41)	Object specified for the text range is not an object that contains text.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.

**Note:** The returned `TextItems` structure references memory that is allocated by the method. Use the `DeallocateTextItems()` method to free this memory when you are done with using it.

### Returns

[TextItems](#)

### Syntax

`GetTextForRange(tr, flags)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>tr</code>	<a href="#">TextRange</a>	No	The text range containing the text you want to get.
<code>flags</code>	<code>int</code>	No	Bit flags that specify the type of text items to retrieve. To get specific types of text items, OR the constants that represent them (for example, <code>FTL_FlowBegin</code> and <code>FTL_String</code> ) into flags. To get all types of text items, specify -1.

## GetTextForRange2

### Description

Gets the text for a specified text range.

You can OR the same flags as in the `GetTextForRange()` method, for the flags parameter.

On success, the method returns a `TextItems` structure containing the array of text items of the requested types.

On failure, the method sets the `len` field of the returned `TextItems` structure to 0, and returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_NotTextObject (-41)	Object specified for the text range is not an object that contains text.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.

**Note:** The returned `TextItems` structure references memory that is allocated by the method. Use the `DeallocateTextItems()` method to free this memory when you are done with using it.

### Returns

`TextItems`

### Syntax

```
GetTextForRange2(tr, flags, flags2)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>tr</code>	<code>TextRange</code>	No	The text range containing the text you want to get.
<code>flags</code>	<code>int</code>	No	Bit flags for the base set text item types that of text items to retrieve.
<code>flags2</code>	<code>int</code>	No	Bit flags for the extended set of text item types that specify the type of text items to retrieve. To get specific types of text items from the extended set, OR the constants that represent them (for example, use a bitwise OR to combine <code>FTI2_RubiTextBegin</code> and <code>FTI2_RubiTextEnd</code> ) into <code>flags2</code> .  To get all types of the extended set of text items, specify -1.

## GetTextProps

### Description

Gets the text properties (such as the format tag, font family and size, and conditions) for a location in text.

As the text properties can be different for each character, you can only get the text properties for an individual location in the text.

On success, the method returns a `PropVals` structure with the text property list for the specified location.

On failure, the method sets the `len` field of the returned `PropVals` structure to 0, and returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.

Error	Reason
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_NotTextObject (-41)	Object specified for the text range is not an object that contains text.
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation

**Returns**

[PropVals](#)

**Syntax**

GetTextProps (textLoc)

**Parameters**

Parameter name	Data Type	Optional	Description
textLoc	<a href="#">TextLoc</a>	No	The text location of the character that you want to get text properties for. The returned properties are the properties that apply to the character to the right of the specified location.

## GetTextPropVal

**Description**

Gets a text property (such as the format tag, font family and size, or conditions) for a location in text.

As a text property can be different for each character, you can get the property for only one location in text at a time.

On success, the method returns a `PropVal` structure with the details of the specified property. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_BadPropNum (-4)	Specified property number is invalid.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation

**Returns**

[PropVal](#)

**Syntax**

GetTextPropVal (textLoc, propNum)

**Parameters**

Parameter name	Data Type	Optional	Description
textLoc	<a href="#">TextLoc</a>	No	The text location of the character for which you want to get the text property. The returned property applies to the character to the right of this location.
propNum	int	No	The property to query. Specify a constant, such as <code>FontFamily</code> .

*Note:* The returned `PropVal` structure references memory that is allocated by the method. Use the `DeallocatePropVal()` method to free this memory when you are done with using it.

**GetTextVal**

**Description**

Gets the value of a specified text property, which can be of any type.

On success, the method returns a `TypedVal` structure containing the value of the specified property. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_BadPropNum (-4)	Specified property number is invalid.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_NotTextObject (-41)	Object specified for the text location is not an object that contains text.
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation

**Returns**

[TypedVal](#)

**Syntax**

`GetTextVal(textLoc, propNum)`

**Parameters**

Parameter name	Data Type	Optional	Description
textLoc	<a href="#">TextLoc</a>	No	The text location of the character for which you want to get the text property. The returned property applies to the character to the right of this location.
propNum	int	No	The property to query. Specify a constant, such as <code>FontFamily</code> .



## HypertextCommand

### Description

Simulates a user-invoked hypertext command.

*Note:* Calling the `HypertextCommand()` method has the same effect as a user clicking on a hypertext marker containing the specified text.

The method returns `FE_Success` on success.

On failure, the method returns the following value to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.

### Returns

int

### Syntax

`HypertextCommand(hypertext)`

### Parameters

Parameter name	Data Type	Optional	Description
hypertext	string	No	A hypertext command to execute, such as <code>gotolink</code> or <code>previouslink</code> . You can specify any command that is valid in a hypertext marker in the document.

## MergeIntoFirst

### Description

Merges the selected structural elements into the first element in the selection.

*Note:* At least two structural elements must be selected in the document before calling the `MergeIntoFirst()` method.

On success, the method does not return any value.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.

### Returns

Void

### Syntax

`MergeIntoFirst()`

## MergeIntoLast

### Description

Merges the selected structural elements into the last element in the selection.

*Note:* At least two structural elements must be selected in the document before calling the `MergeIntoLast()` method.

On success, the method does not return any value.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.

### Returns

Void

### Syntax

```
MergeIntoLast ()
```

## NewTable

### Description

Inserts a table (`FO_Tbl` object).

When you create a table in the user interface, you can specify a Table Catalog format for the table. FrameMaker uses the following properties of the Table Catalog format as the defaults for the new table:

- Number of body rows (`TblInitNumBodyRows`)
- Number of columns (`TblInitNumCols`)
- Number of footer rows (`TblInitNumFRows`)
- Number of header rows (`TblInitNumHRows`)
- Paragraph formats for header, body, and footer cells

For example, if the Table Catalog format's `TblInitNumCols` property is set to 8, the `NumCols` property of the new table is set to 8. With the `NewTable()` method, you can use the Table Catalog format properties as defaults for the number of rows and columns in a new table, or you can provide your own defaults. After you have create a table, you can add or remove rows with the `AddRows()` and `DeleteRows()` methods. You can add or remove columns with the `AddCols()` and `DeleteCols()` methods. If you use the `NewTable()` method to create a table in a structured FrameMaker document, FrameMaker applies default element tags, such as `Table`, `Row`, and `Cell`, to the table element and its child elements. To make these elements valid, you must add code to change their tags. In most cases it is easier to add tables to structured documents by calling the `NewElementInHierarchy()` or `NewElement()` methods to add a table element.

On success, the method returns the newly added table.

On failure, the method returns one of the following values to `FA_errno`:

**Function Summary**

<b>Error</b>	<b>Reason</b>
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_NotTextObject (-41)	Object specified for the text location is not a paragraph (FO_Pgf).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.
Constants.FE_BadOperation (-27)	Function call specified an illegal operation.
Constants.FE_BadNew (-23)	Table cannot be created; the format specified by <code>format</code> may not exist or the text location specified by <code>textLoc</code> is in a table or a footnote.

**Returns**

[Tbl](#)

**Syntax**

```
NewTable(format, numCols, numBodyRows, numHeaderRows, numFooterRows, textLoc)
```

**Parameters**

<b>Parameter name</b>	<b>Data Type</b>	<b>Optional</b>	<b>Description</b>
format	string	No	The table format tag (for example, FormatA or Wide Table). To use the default format, specify NULL.
numCols	int	No	The number of columns in the table. To use the default number of columns from the Table Catalog format, specify -1.
numBodyRows	int	No	The number of rows in the table. To use the default number of body rows from the Table Catalog format, specify -1.
numHeaderRows	int	No	The number of heading rows in the table. To use the default number of header rows from the Table Catalog format, specify -1.
numFooterRows	int	No	The number of footing rows in the table. To use the default number of footer rows from the Table Catalog format, specify -1.
textLoc	<a href="#">TextLoc</a>	No	The location at which to insert the new table. The location cannot be within a footnote or a table.

**Paste****Description**

Pastes the contents of the FrameMaker Clipboard into a specified document at the insertion point. Cutting and Pasting objects causes FrameMaker to create an ID for the pasted object.

Specifying a value of 0 for the flags parameter, causes the Paste() method to suppress any interactive alerts or warnings that arise. In addition, the method inserts columns to the left of the current columns and rows above the current row.

You can OR the following values into flags:

<b>Flags</b>	<b>Meaning</b>
Constants.FF_INTERACTIVE (0x0001)	Prompt the user with dialog or alert boxes that arise.

**Function Summary**

Flags	Meaning
Constants.FF_VISIBLE_ONLY (0x0010)	Cut only the visible portion of the selection.
Constants.FF_DONT_DELETE_HIDDEN_TEXT (0x0004)	Do not replace hidden text.
Constants.FF_DONT_APPLY_ALL_ROWS (0x0040)	Do no't apply condition setting on the Clipboard to all rows. If whole table is selected and the Clipboard contains condition setting, cancel the paste operation.
Constants.FF_REPLACE_CELLS (0x0020)	Replace selected cells with cells on the Clipboard.
Constants.FF_INSERT_BELOW_RIGHT (0x0008)	Add columns to the right of the current column or below the current row.

When you use the `Paste()` method to paste table cells into a table, it does not work exactly like the interactive Paste command.

The interactive Paste command automatically overwrites cells if the Clipboard contains less than an entire row or column. For example, if the insertion point is in a three-column table and the Clipboard contains a single cell, the interactive Paste command overwrites the cell containing the insertion point with the cell on the Clipboard.

If two cells in the table are selected, the interactive Paste command overwrites both of them with the cell on the Clipboard.

By default, the `Paste()` method does not overwrite any cells. If the Clipboard contains less than an entire row or column when you call `Paste()`, or if the current selection is less than an entire row, the `Paste()` method does nothing and returns `Constants.FE_BadSelectionForOperation (-59)`. The method ensures that you do not inadvertently overwrite any cells.

To make the `Paste()` method replace cells with the Clipboard contents, you must call it with the `Constants.FF_REPLACE_CELLS (0x0020)` flag set. The `Constants.FF_INTERACTIVE (0x0001)` flag takes precedence over other flags.

If you specify `Constants.FF_INTERACTIVE (0x0001) | Constants.FF_DONT_DELETE_HIDDEN_TEXT (0x0004)` and the selection contains hidden text, FrameMaker prompts the user, and allows the user to choose whether to delete the hidden text or retain it.

It is illegal to specify `Constants.FF_REPLACE_CELLS (0x0020) | Constants.FF_INSERT_BELOW_RIGHT (0x0008)`.

The method returns `FE_Success` on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadOperation (-27)	Function call specified an illegal operation.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.
Constants.FE_Canceled (-44)	User canceled the operation.

**Returns**

`int`

**Syntax**

`Paste(flags)`

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Bit field that specifies how to paste the text and how to handle interactive alerts. For default settings, specify 0.

## PromoteElement

### Description

Promotes the selected structural element. The selected element becomes a sibling of its former parent and appears immediately after its former parent. The siblings that follow it become the element's children.

**Note:** One structural element must be selected when the `PromoteElement()` method is called. The selected element cannot be the root element or a child of the root element.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	The current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.

### Returns

Void

### Syntax

```
PromoteElement ()
```

## QuickSelect

### Description

Implements a quick-key interface that allows the user to choose a string from a list of strings in the document Tag area.

The `QuickSelect()` method highlights the document Tag area and displays a prompt and the first string in a specified list of strings.

The user can display a string in the Tag area by typing the first few letters of the string. The user can also scroll through the strings by pressing the up and down arrow keys.

To choose a string, the user presses Return when the string appears in the Tag area. To cancel the choice, the user clicks in the document without pressing Return.

On success, the method returns an index into the array of strings specified by `stringlist` or -1 if the user cancels the quick selection.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.
Constants.FE_BadDocId (-2)	Invalid document.

### Returns

int

### Syntax

```
QuickSelect(prompt, stringlist)
```

### Parameters

Parameter name	Data Type	Optional	Description
prompt	string	No	The prompt that appears in the Tag area
stringlist	Strings	No	The list of strings from which the user can choose

## Redisplay

### Description

Updates the display for a specified document to reflect any changes that occurred when the `Displaying` property was set to `False`. If you have set the `Displaying` property to `False` and subsequently reset it to `True`, you should call the `Redisplay()` method to redisplay each document you modified.

The method returns `FE_Success` on success. On failure, the method returns the following value to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.

### Returns

int

### Syntax

```
Redisplay()
```

## Reformat

### Description

Reformats the specified document. If you have disabled and subsequently re-enabled reformatting by setting the session property, `Reformatting` to `True`, you should call the `Reformat()` method to reformat each changed document in the session.

The method returns `FE_Success` on success. On failure, the method returns the following value to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.

### Returns

int

### Syntax

Reformat ()

## Rehyphenate

### Description

Rehyphenates a specified document based on changes the user has made to words' hyphenation points.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.
Constants.FE_SystemError (-42)	Could not allocate memory.

### Returns

int

### Syntax

Rehyphenate ()

## ResetEqnSettings

### Description

Resets the document equation settings to the default settings.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.
Constants.FE_SystemError (-42)	Could not allocate memory.

### Returns

int

### Syntax

ResetEqnSettings ()

## ResetReferenceFrames

### Description

Resets the reference frames in the specified document. This method is useful for updating a document after you have programmatically changed a reference frame that is referenced by paragraphs in the document.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker version does not support this operation.
<code>Constants.FE_SystemError (-42)</code>	Could not allocate memory.

### Returns

`int`

### Syntax

```
ResetReferenceFrames()
```

## RestartPgfnNumbering

### Description

Restarts the paragraph numbering for a specified document.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker version does not support this operation.
<code>Constants.FE_SystemError (-42)</code>	Could not allocate memory.

### Returns

`int`

### Syntax

```
RestartPgfnNumbering()
```

## ScrollToText

### Description

Scrolls the document window to a specified text range. The method scrolls the document to the end of the range that is closest to the current display position.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:



Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid Object
Constants.FE_NotTextObject (-41)	Object specified for the text location is not a paragraph (FO_Pgf) or a flow (FO_Flow).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text range could not be found in the specified paragraph or text line.
Constants.FE_BadRange (-36)	The specified text range is invalid.

**Returns**

int

**Syntax**

ScrollToText (textRange)

**Parameters**

Parameter name	Data Type	Optional	Description
textRange	TextRange	No	The text range to which to scroll.

**SetTextProps**

**Description**

Sets the text properties (such as the format tag, font family, and size) for a text range.

On success, the method does not return anything.

On failure, the method returns one of the following values to FA\_errno:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadName (-33)	Specified name is illegal.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_GenRuleAmbiguous (-55)	General rule in structured document is ambiguous.
Constants.FE_GenRuleConnectorExpected (-57)	General rule in structured document is missing a connector.
Constants.FE_GenRuleItemExpected (-51)	General rule in structured document is missing a rule item.
Constants.FE_GenRuleLeftBracketExpected (-53)	General rule in structured document is missing a left bracket.
Constants.FE_GenRuleMixedConnectors (-52)	General rule in structured document has mixed connectors.
Constants.FE_GenRuleRightBracketExpected (-54)	General rule in structured document is missing a right bracket.
Constants.FE_GenRuleSyntaxError(-56)	General rule in structured document has a syntax error.
Constants.FE_NotTextObject (-41)	Object specified for the text range is not a paragraph (FO_Pgf) or a flow (FO_Flow).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text range could not be found in the specified paragraph or text line.

Error	Reason
Constants.FE_OutOfRange (-7)	Specified property value is out of the legal range for the specified property.
Constants.FE_ReadOnly (-6)	Property is read-only and cannot be set.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.

**Returns**

Void

**Syntax**

SetTextProps (textRange, setVal)

**Parameters**

Parameter name	Data Type	Optional	Description
textRange	TextRange	No	The text range to which to apply the property list.
setVal	PropVals	No	The property list to apply to the text range.

**SetTextPropVal**

**Description**

Sets a text property for a specified text range. The property can be of any type.

On success, the method does not return anything.

On failure, the method returns one of the following values to FA\_errno:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadName (-33)	Specified name is illegal.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_BadPropNum (-4)	Specified property number is invalid.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_GenRuleAmbiguous (-55)	General rule in structured document is ambiguous.
Constants.FE_GenRuleConnectorExpected (-57)	General rule in structured document is missing a connector.
Constants.FE_GenRuleItemExpected (-51)	General rule in structured document is missing a rule item.
Constants.FE_GenRuleLeftBracketExpected (-53)	General rule in structured document is missing a left bracket.
Constants.FE_GenRuleMixedConnectors (-52)	General rule in structured document has mixed connectors.
Constants.FE_GenRuleRightBracketExpected (-54)	General rule in structured document is missing a right bracket.
Constants.FE_GenRuleSyntaxError(-56)	General rule in structured document has a syntax error.
Constants.FE_NotTextObject (-41)	Object specified for the text range is not a paragraph (FO_Pgf) or a flow (FO_Flow).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text range could not be found in the specified paragraph or text line.

Error	Reason
Constants.FE_OutOfRange (-7)	Specified property value is out of the legal range for the specified property.
Constants.FE_ReadOnly (-6)	Property is read-only and cannot be set.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.

**Returns**

Void

**Syntax**

SetTextPropVal (textRange, setVal)

**Parameters**

Parameter name	Data Type	Optional	Description
textRange	TextRange	No	The text range to which the property is to be applied.
setVal	PropVal	No	The property to apply to the text range.

**SetTextVal**

**Description**

Sets a specified text property for a text range.

On success, the method does not return anything.

On failure, the method returns one of the following values to FA\_errno:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadName (-33)	Specified name is illegal.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_BadPropNum (-4)	Specified property number is invalid.
Constants.FE_BadRange (-36)	Specified text range is invalid.
Constants.FE_GenRuleAmbiguous (-55)	General rule in structured document is ambiguous.
Constants.FE_GenRuleConnectorExpected (-57)	General rule in structured document is missing a connector.
Constants.FE_GenRuleItemExpected (-51)	General rule in structured document is missing a rule item.
Constants.FE_GenRuleLeftBracketExpected (-53)	General rule in structured document is missing a left bracket.
Constants.FE_GenRuleMixedConnectors (-52)	General rule in structured document has mixed connectors.
Constants.FE_GenRuleRightBracketExpected (-54)	General rule in structured document is missing a right bracket.
Constants.FE_GenRuleSyntaxError(-56)	General rule in structured document has a syntax error.
Constants.FE_NotTextObject (-41)	Object specified for the text range is not a paragraph (FO_Pgf) or a flow (FO_Flow).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text range could not be found in the specified paragraph or text line.

Error	Reason
Constants.FE_OutOfRange (-7)	Specified property value is out of the legal range for the specified property.
Constants.FE_ReadOnly (-6)	Property is read-only and cannot be set.
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.

**Returns**

Void

**Syntax**

`SetTextVal(textRange, propNum, setVal)`

**Parameters**

Parameter name	Data Type	Optional	Description
textRange	<a href="#">TextRange</a>	No	The text range to which the property is to be applied.
propNum	int	No	The number of the property to set.
setVal	<a href="#">TypedVal</a>	No	The value of the property to set.

**SilentPrintDoc**

**Description**

Prints a document or a book using the default print settings. Default print settings are the settings that appear in the Print dialog box when the user attempts to print a document.

The `SilentPrintDoc()` method initializes the print page size and printer name if they do not have values.

To change a document's default print settings, set the document's print properties. [To be linked to the print properties]

For example, to change the turn registration marks on when you print a document, use the `SetInt()` method to set the document's `RegistrationMarks` property.

If you save the document or attempt to print it again within the same session, any changes you make to a document's print settings (except `PrintStartPage` and `PrintEndPage`) appears the next time the user displays the Print dialog box for the document.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_SystemError (-42)	Could not open or close the printer file.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.

**Returns**

int

**Syntax**

`SilentPrintDoc()`

## SimpleImportFormats

### Description

Imports formats from a document to a document or a book. If you import formats to a book, the method imports formats to each book component for which the `ImportFmtInclude` property is set to `True`.

You can OR the following values into the `formatFlags` parameter to specify which formats to import.

Value	Format to Import
Constants.FF_UFF_COLOR (0x0100)	Colors
Constants.FF_UFF_COMBINED_FONTS (0x0800)	Combined Fonts
Constants.FF_UFF_COND (0x0010)	Conditions
Constants.FF_UFF_DOCUMENT_PROPS (0x0400)	Document Properties
Constants.FF_UFF_FONT (0x0002)	Character Catalog Formats
Constants.FF_UFF_MATH (0x0200)	Equation Settings
Constants.FF_UFF_PAGE (0x0004)	Page Layouts
Constants.FF_UFF_PGF (0x0001)	Paragraph Catalog Formats
Constants.FF_UFF_REFPAGE (0x0020)	Reference Pages
Constants.FF_UFF_REMOVE_EXCEPTIONS (0x8000)	Remove exception formats from target documents
Constants.FF_UFF_REMOVE_PAGE_BREAKS (0x4000)	Remove all forced page breaks from target documents
Constants.FF_UFF_TABLE (0x0008)	Table Catalog Formats
Constants.FF_UFF_VAR (0x0040)	Variable Formats
Constants.FF_UFF_XREF (0x0080)	Cross Reference Formats

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support this operation.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_Canceled (-44)	User canceled the operation.
Constants.FE_FailedState (-45)	The FrameMaker product failed to open one or more of the book's document files during the import operation.

### Returns

int

### Syntax

`SimpleImportFormats (fromDoc, formatFlags)`

**Parameters**

Parameter name	Data Type	Optional	Description
fromDoc	Doc	No	The document or book from which to import formats.
formatFlags	int	No	Bit field specifying which formats to import. Specify 0 for the default flags.

You can OR the following values into the `formatFlags` parameter to specify which formats to import.

Value	Import
Constants.FF_UFF_COLOR (0x100)	Import colors.
Constants.FF_UFF_COMBINED_FONTS (0x800)	Import combined fonts.
Constants.FF_UFF_COND (0x0010)	Import conditions.
Constants.FF_UFF_DOCUMENT_PROPS (0x0400)	Import document properties.
Constants.FF_UFF_FONT (0x0002)	Import Character Catalog formats.
Constants.FF_UFF_MATH (0x200)	Import equation settings.
Constants.FF_UFF_PAGE (0x0004)	Import page layouts.
Constants.FF_UFF_PGF (0x0001)	Import Paragraph Catalog formats.
Constants.FF_UFF_REFPAGE (0x0020)	Import reference pages.
Constants.FF_UFF_REMOVE_EXCEPTIONS (0x8000)	Remove exception formats from target documents.
Constants.FF_UFF_REMOVE_PAGE_BREAKS (0x4000)	Remove all forced page breaks from target documents.
Constants.FF_UFF_TABLE (0x0008)	Import Table Catalog formats.
Constants.FF_UFF_VAR (0x0040)	Import variable formats.
Constants.FF_UFF_XREF (0x0080)	Import cross reference formats.

**SplitElement**

**Description**

Splits the structural element containing the insertion point into two elements at the insertion point. The insertion point must be inside the element you want to split.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.
Constants.FE_WrongProduct (-60)	Current FrameMaker interface is not Structured FrameMaker.

**Returns**

Void

### Syntax

`SplitElement()`

## TextLocToElementLoc

### Description

Returns the element location structure that corresponds to the current text location.

On success, the method returns an `ElementLoc` structure containing the element location.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadParameter ()	<code>tloc</code> parameter was empty or was improperly specified
Constants.FE_WrongProduct (-60)	Current FrameMaker interface is not Structured FrameMaker.

### Returns

[ElementLoc](#)

### Syntax

`TextLocToElementLoc(tloc)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>tloc</code>	<a href="#">TextLoc</a>	No	The text location structure to convert.

## UnWrapElement

### Description

Removes the selected structural elements, but leaves their contents and child elements intact in the document.

The `UnWrapElement()` method does not remove all the elements in the selection but only removes the top-level elements.

**Note:** At least one structural element must be selected before calling the `UnWrapElement()` method.

**Note:** The `UnWrapElement()` method has no effect on object elements.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.
Constants.FE_WrongProduct (-60)	Current FrameMaker interface is not Structured FrameMaker.

### Returns

Void

### Syntax

UnWrapElement ()

## UpdateXRefs

### Description

Updates the cross-references in a document. The method performs the same operation as clicking Update in the Cross-Reference window.

You can OR the following values into the `updateXRefFlags` parameter:

Flags	Meaning
Constants.FF_XRUI_FORCE_UPDATE (0x08)	Updates all cross-references, regardless of whether the source document has changed
Constants.FF_XRUI_INTERNAL (0x01)	Only update internal cross-references
Constants.FF_XRUI_OPEN_DOCS (0x02)	Only update cross-references whose sources are in open documents
Constants.FF_XRUI_CLOSED_DOCS (0x04)	Only update cross-references whose sources are in closed documents
Constants.FF_XRUI EVERYTHING (0x01 0x02 0x03)	Update all cross references.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document
Constants.FE_SomeUnresolved (-22)	There are some unresolved references after the update

### Returns

int

### Syntax

UpdateXRefs (updateXRefFlags)

### Parameters

Parameter name	Data Type	Optional	Description
updateXRefFlags	int	No	Flags that indicate which cross-references to update.

## UpdateDITAReferences

### Description

Updates all DITA references of the specified type.



**Returns**

void

**Syntax**

UpdateDITAResferences(docId, flag);

**Parameters**

Parameter name	Data Type	Optional	Description
docId	int	No	The ID of the document containing the inset.
flag	int	No	The available flags and their values are as follows:  FF_DITAUpdateAllConrefs: 0x01  FF_DITAUpdateAllXrefs: 0x02  FF_DITAUpdateAllLinks: 0x04  FF_DITAUpdateAllTopicrefs: 0x08  FF_DITAUpdateAllTopicsetrefs: 0x10  FF_DITAUpdateAllReferences: FF_DITAUpdateAllConrefs   FF_DITAUpdateAllXrefs   FF_DITAUpdateAllLinks   FF_DITAUpdateAllTopicrefs   FF_DITAUpdateAllTopicsetrefs

If UpdateDITAResferences() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_WrongProduct	Current FrameMaker product doesn't support the operation.
Constants.FE_BadDocId	The Document ID provided is invalid.
Constants.FE_NonDITADocument	The Document provided is not a DITA document.

**SimpleImportElementDefs**

**Description**

Imports element definitions and the format change list catalog from an EDD or FrameMaker document or book to a FrameMaker document or book.

*Note: If you import element definitions to a book, the SimpleImportElementDefs() method imports element definitions to each book component for which the ImportFmtInclude property is set to True.*

You can OR the following flags into the importFlags parameter:

Flag	Meaning
Constants.FF_IED_REMOVE_OVERRIDES (0x0001)	Clear format overrides
Constants.FF_IED_REMOVE_BOOK_INFO (0x0002)	Clear formatting inherited from a parent book, if the method is called on a document
Constants.FF_IED_DO_NOT_IMPORT_EDD (0x0004)	Do not treat the target document as an EDD but just import its element catalog, if the target document is an EDD

Flag	Meaning
Constants.FF_IED_NO_NOTIFY (0x0008)	Do not issue the Constants.FA_Note_PreImportElemDefs (64) or Constants.FA_Note_PostImportElemDefs (65) notifications.

The method returns `FE_Success` on success. On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the requested operation
Constants.FE_BadDocId (-2)	Invalid document

**Returns**

int

**Syntax**

`SimpleImportElementDefs (fromDocOrBook, importFlags)`

**Parameters**

Parameter name	Data Type	Optional	Description
fromDocOrBook	Doc   Book	No	The document or book from which the element definitions are to be imported
importFlags	int	No	Specifies how to import formats

## GetUniqueObject

**Description**

Gets the object from its persistent unique identifier (UID).

FrameMaker assigns a UID to each object in a document or book that is not identified by a name. The UID, unlike the object's ID, does not change from one session to another.

*Note: When you copy an object to the clipboard and paste it, FrameMaker changes the UID. This also happens when you hide and show conditional text.*

The method returns the object, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document
Constants.FE_TypeUnNamed (-19)	Objects of the specifies type are not identified by UIDs
Constants.FE_NameNotFound (-20)	Objects with the specified UID could not be found

**Returns**

Object

**Syntax**

GetUniqueObject(objType, unique)

**Parameters**

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object (for example, FO_Pgf)
unique	int	No	UID of the object

**Close**

**Description**

Closes a document or a book.

*Note: If there are unsaved changes in a document and you set Constants.FF\_CLOSE\_MODIFIED (1) for the flags parameter, then the Close() method abandons the changes and closes the file anyway.*

*Note: If you set the flags parameter to 0, the Close() method aborts the Close operation and returns Constants.FE\_DocModified (-8).*

The method returns FE\_Success, on success.

On failure, the method returns the following value to FA\_errno:

Error	Reason
Constants.FE_DocModified (-8)	The document was modified and the flags parameter was set to 0

**Returns**

int

**Syntax**

Close(flags)

**Parameters**

Parameter name	Data Type	Optional	Description
flags	int	No	Specifies whether to abort or close open documents or books if they have unsaved changes. Set the Constants.FF_CLOSE_MODIFIED (1) flag to close open documents and books regardless of their state.

**SimpleSave**

**Description**

Saves a document or book.

If you set the interactive parameter to False and specify the document or book's current name for saveAsName, FrameMaker saves the document or book under its current name.

If you specify another filename for saveAsName, FrameMaker saves the document or book to that filename.

**Function Summary**

If you specify an empty string (""), FrameMaker does not save the file. Instead it sets `FA_errno` to `Constants.FE_BadParameter (-43)`. If you set the `interactive` parameter to `True`, FrameMaker displays the Save dialog box and allows the user to choose a filename. The document or book's current name appears as the default name.

The method returns the document it saved, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<ul style="list-style-type: none"> <li>Constants.FE_Canceled (-44)</li> <li>Constants.FE_CanceledByClient (-86)</li> </ul>	<ul style="list-style-type: none"> <li>Constants.FV_FileNotWritable (33): File was not writable.</li> <li>Constants.FV_BadSaveFileName (34): The specified file name is not allowed by the operating system.</li> <li>Constants.FV_BadFileId (35): The file's operating system ID was bad.</li> <li>Constants.FV_CancelSaveFilesInUse (50): The file is in use and the user did not or could not reset the lock. Or the file is in use, and the <code>Constants.FS_FileIsInUse (17)</code> parameter is set to <code>Constants.FV_DoCancel (0)</code>, or it is set to <code>Constants.FV_ResetLockAndContinue (7)</code> but FrameMaker could not reset the lock.</li> <li>Constants.FV_CancelSaveModDateChanged (52): The file has changed since the last time it was opened or saved in the current session.</li> <li>Constants.FV_LockCouldntBeReset (2): File lock could not be reset.</li> <li>Constants.FV_LockWasReset (0): File lock was reset.</li> <li>Constants.FV_LockNotReset (1): File lock was not reset.</li> <li>Constants.FV_FilesViewOnly (4): File was View Only.</li> <li>Constants.FV_NonPortableSaveName (37): File had a name that is not portable (Macintosh only).</li> <li>Constants.FV_NonPortableFileRefs (38): Referenced graphics files had filenames that are not portable (Macintosh only).</li> </ul>
Constants.FE_WrongProduct (-60)	The specified file contains structure and the current product interface is not Structured FrameMaker.
Constants.FE_FailedState (-45)	The filename was invalid.
Constants.FE_BadParameter (-43)	
Constants.FE_FilterFailed (-89)	Constants.FV_InvalidSaveFilter (42): The filter specified by <code>Constants.FS_SaveFileTypeHint (16)</code> is not installed, or the syntax for <code>Constants.FS_SaveFileTypeHint (16)</code> is invalid.

**Returns**

[Doc](#)

**Syntax**

```
SimpleSave(saveAsName, interactive)
```

**Parameters**

Parameter name	Data Type	Optional	Description
saveAsName	string	No	The absolute path in which to save the document or book.
interactive	int	No	Specifies whether the FrameMaker product displays messages and warnings to the user. Setting this to True displays messages and warnings.

**NewGraphicObject**

**Description**

Creates the following types of graphic objects:

- Arc
- Ellipse
- Flow
- Group
- Inset
- Line
- Math
- Polyline
- Polygon
- Rectangle
- RoundRect
- TextFrame
- TextLine
- UnanchoredFrame

*Note:* To create an anchored frame, use `NewAnchoredObject()`.

If there is more than one object within the parent frame, the `NewGraphicObject()` method adds the new ESTK graphic object to the end of the linked list of child objects. That is, it puts it in the front of the back-to-front draw order. The method automatically takes care of updating the object's `PrevGraphicInFrame` and `NextGraphicInFrame` properties. The `NewGraphicObject()` method gives the new method graphic object a set of arbitrary default properties.

The method returns the newly created graphic object, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_NotFrame (-12)	Specified parent object is not a frame.
Constants.FE_BadNew (-23)	Object cannot be created.

### Returns

Object

### Syntax

```
NewGraphicObject(objType, parent)
```

### Parameters

Parameter name	Data Type	Optional	Description
objType	int	No	The type of ESTK graphic object to create (for example, FO_Rectangle or FO_Line).
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the object.

## NewArc

### Description

Creates an arc object.

See [NewGraphicObject](#) for more information.

### Returns

[Arc](#)

### Syntax

```
NewArc(parent)
```

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the arc.

## NewEllipse

### Description

Creates an ellipse object.

See [NewGraphicObject](#) for more information.

### Returns

[Ellipse](#)

### Syntax

```
NewEllipse(parent)
```

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the ellipse.

## NewFlow

### Description

Creates a flow object.

See [NewGraphicObject](#) for more information.

### Returns

[Flow](#)

### Syntax

`NewFlow (parent)`

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the flow.

## NewGroup

### Description

Creates a group object.

See [NewGraphicObject](#) for more information.

### Returns

[Group](#)

### Syntax

`NewGroup (parent)`

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the group.

## NewInset

### Description

Creates an inset object.

See [NewGraphicObject](#) for more information.

### Returns

[Inset](#)

### Syntax

`NewInset (parent)`

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the inset.

## NewLine

### Description

Creates a line object.

See [NewGraphicObject](#) for more information.

### Returns

[Line](#)

### Syntax

`NewLine (parent)`

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the line.

## NewMath

### Description

Creates a math object.

See [NewGraphicObject](#) for more information.



**Function Summary****Returns**[Math](#)**Syntax**

NewMath (parent)

**Parameters**

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the math object.

**NewMathML****Description**

Creates a MathML object.

See [NewGraphicObject](#) for more information.**Returns**[MathML](#)**Syntax**

NewMathML (parent)

**Parameters**

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the MathML object.

**NewPolyline****Description**

Creates a polyline object.

See [NewGraphicObject](#) for more information.**Returns**[Polyline](#)**Syntax**

NewPolyline (parent)

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the polyline.

## NewPolygon

### Description

Creates a polygon object.

See [NewGraphicObject](#) for more information.

### Returns

[Polygon](#)

### Syntax

`NewPolygon (parent)`

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the polygon.

## NewRectangle

### Description

Creates a rectangle object.

See [NewGraphicObject](#) for more information.

### Returns

[Rectangle](#)

### Syntax

`NewRectangle (parent)`

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the rectangle.

## NewRoundRect

### Description

Creates a round rectangle object.

See [NewGraphicObject](#) for more information.

### Returns

[RoundRect](#)

### Syntax

`NewRoundRect (parent)`

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the round rectangle.

## NewTextFrame

### Description

Creates a text frame object.

See [NewGraphicObject](#) for more information.

### Returns

[TextFrame](#)

### Syntax

`NewTextFrame (parent)`

### Parameters

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the text frame.

## NewTextLine

### Description

Creates a text line object.

See [NewGraphicObject](#) for more information.

**Function Summary****Returns**[TextLine](#)**Syntax**

NewTextLine (parent)

**Parameters**

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the text line.

**NewUnanchoredFrame****Description**

Creates an unanchored frame object.

See [NewGraphicObject](#) for more information.**Returns**[UnanchoredFrame](#)**Syntax**

NewUnanchoredFrame (parent)

**Parameters**

Parameter name	Data Type	Optional	Description
parent	Any FrameMaker Graphic Object	No	The parent frame in which to create the unanchored frame.

**NewAnchoredFormattedObject****Description**

Creates the following types of anchored objects:

- Var
- XRef
- Tbl

`NewAnchoredFormattedObject ()` inserts the object at the specified location in text. It uses arbitrary default properties for the new object.

If you call `NewAnchoredFormattedObject ()` to create a table, it uses the default numbers of rows and columns from the specified Table Catalog format. To use the default Table Catalog format for a new table, set the format to `Null`. To specify the number of rows and columns when you create a table, use `NewTable ()`.

`NewAnchoredFormattedObject()` returns the created anchor object, on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_NotTextObject (-41)	Object specified for text location is not a paragraph (FO_Pgf).
Constants.FE_OffsetNotFound(-21)	Offset specified for the text location could not be found in the specified paragraph or text line.
Constants.FE_BadNew(-23)	Object cannot be created; the format specified by <code>format</code> may not exist.

### Returns

Object

### Syntax

`NewAnchoredFormattedObject(objType, format, textLoc)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>objType</code>	<code>int</code>	No	The type of object to create (for example, <code>FO_XRef</code> ).
<code>format</code>	<code>string</code>	No	The string that specifies the object's format (for example, <code>Heading &amp; Page</code> for a cross-reference, <code>Format A</code> for a table, or <code>Current Date (Long)</code> for a variable).
<code>textLoc</code>	<code>TextLoc</code>	No	The text location at which to insert the anchored object.

## NewAnchoredFormattedVar

### Description

Creates an `AnchoredFormattedVar` object.

See [NewAnchoredFormattedObject](#) under the `Doc` class for more information.

### Returns

`Var`

### Syntax

`NewAnchoredFormattedVar(format, textLoc)`

### Parameters

Parameter name	Data Type	Optional	Description
format	string	No	The string that specifies the object's format (for example, <code>Heading &amp; Page</code> for a cross-reference, <code>Format A</code> for a table, or <code>Current Date (Long)</code> for a variable).
textLoc	<a href="#">TextLoc</a>	No	The text location at which to insert the <code>AnchoredFormattedVar</code> object.

## NewAnchoredFormattedXRef

### Description

Creates an `AnchoredFormattedXRef` object.

See [NewAnchoredFormattedObject](#) under the Doc class for more information.

### Returns

[XRef](#)

### Syntax

```
NewAnchoredFormattedXRef(format, textLoc)
```

### Parameters

Parameter name	Data Type	Optional	Description
format	string	No	The string that specifies the object's format (for example, <code>Heading &amp; Page</code> for a cross-reference, <code>Format A</code> for a table, or <code>Current Date (Long)</code> for a variable).
textLoc	<a href="#">TextLoc</a>	No	The text location at which to insert the <code>AnchoredFormattedXRef</code> object.

## NewAnchoredFormattedTbl

### Description

Creates an `AnchoredFormattedTbl` object.

See [NewAnchoredFormattedObject](#) under the Doc class for more information.

### Returns

[Tbl](#)

### Syntax

```
NewAnchoredFormattedTbl(format, textLoc)
```

**Parameters**

Parameter name	Data Type	Optional	Description
format	string	No	The string that specifies the object's format (for example, <code>Heading &amp; Page</code> for a cross-reference, <code>Format A</code> for a table, or <code>Current Date (Long)</code> for a variable).
textLoc	<code>TextLoc</code>	No	The text location at which to insert the <code>AnchoredFormattedTbl</code> object.

**NewAnchoredObject**

**Description**

Creates any of the following anchored objects:

- AFrame
- Fn
- Marker
- TiApiClient
- Table

`NewAnchoredObject ()` inserts the object at the specified location in text. It uses arbitrary default properties for the new object. Tables created by `NewAnchoredObject ()` have a single column and a single body row. It is usually easier to use `NewTable ()` to create tables.

The method returns the created anchored object on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId(-2)</code>	Invalid document.
<code>Constants.FE_BadNew(-23)</code>	Object cannot be created.
<code>Constants.FE_BadObjId(-3)</code>	Invalid object.
<code>Constants.FE_BadOperation (-27)</code>	Method call specified a bad operation.
<code>Constants.FE_NotTextObject (-41)</code>	Object specified for text location is not a paragraph ( <code>FO_Pgf</code> ).
<code>Constants.FE_OffsetNotFound(-21)</code>	Offset specified for the text location could not be found in the specified paragraph or text line.

**Returns**

Object

**Syntax**

`NewAnchoredObject (objType, textLoc)`

### Parameters

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object to create (for example, FO_Marker or FO_Fn).
textloc	<a href="#">TextLoc</a>	No	The text location at which to insert the anchored object.

## NewAnchoredAFrame

### Description

Creates an anchored AFrame object.

See [NewAnchoredObject](#) under the Doc class for more information.

### Returns

[AFrame](#)

### Syntax

```
NewAnchoredAFrame(textloc)
```

### Parameters

Parameter name	Data Type	Optional	Description
textloc	<a href="#">TextLoc</a>	No	The text location at which to insert the anchored AFrame.

## NewAnchoredFn

### Description

Creates an anchored Fn object.

See [NewAnchoredObject](#) under the Doc class for more information.

### Returns

[Fn](#)

### Syntax

```
NewAnchoredFn(textloc)
```

### Parameters

Parameter name	Data Type	Optional	Description
textloc	<a href="#">TextLoc</a>	No	The text location at which to insert the anchored Fn object.

## NewAnchoredMarker

### Description

Creates an anchored Marker object.



See [NewAnchoredObject](#) under the Doc class for more information.

### Returns

[Marker](#)

### Syntax

`NewAnchoredMarker (textloc)`

### Parameters

Parameter name	Data Type	Optional	Description
textloc	<a href="#">TextLoc</a>	No	The text location at which to insert the anchored Marker.

## NewAnchoredTiApiClient

### Description

Creates an anchored TiApiClient object.

See [NewAnchoredObject](#) under the Doc class for more information.

### Returns

[TiApiClient](#)

### Syntax

`NewAnchoredTiApiClient (textloc)`

### Parameters

Parameter name	Data Type	Optional	Description
textloc	<a href="#">TextLoc</a>	No	The text location at which to insert the anchored TiApiClient object.

## NewAnchoredTbl

### Description

Creates an anchored table object.

See [NewAnchoredObject](#) under the Doc class for more information.

### Returns

[Tbl](#)

### Syntax

`NewAnchoredTbl (textloc)`

**Parameters**

Parameter name	Data Type	Optional	Description
textloc	TextLoc	No	The text location at which to insert the anchored Table.

**NewSeriesObject**

**Description**

Creates the following series objects:

- BodyPage
- BookComponent
- Pgf

The method allows you to specify the position in the series at which to add the new object.

The method returns the created series object, on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadNew (-23)	Object cannot be created.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_NotBodyPage (-24)	The <code>prevId</code> parameter must specify a body page.
Constants.FE_NotBodyPgf (-25)	The <code>prevId</code> parameter must specify a paragraph..
Constants.FE_NotBookComponent (-26)	The <code>prevId</code> parameter must specify a book component..

**Returns**

One of the following objects: Body page, Pgf and Book Component

**Syntax**

`NewSeriesObject(objType, previous)`

**Parameters**

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object to create (for example, <code>FO_BodyPage</code> or <code>FO_Pgf</code> ).
previous	One of: Body page, Pgf and Book Component	No	The object in the series after which to add the new object. To add a paragraph at the start of a flow, specify the flow. To add an object at the beginning of any other series, specify 0.

## NewSeriesBodyPage

### Description

Creates a body page.

See [NewSeriesObject](#) under the Doc class for more information.

### Returns

[BodyPage](#)

### Syntax

`NewSeriesBodyPage (previous)`

### Parameters

Parameter name	Data Type	Optional	Description
previous	BodyPage	No	The object in the series after which to add the new body page. To add a paragraph at the start of a flow, specify the flow. To add a body page at the beginning of any other series, specify 0.

## NewSeriesPgf

### Description

Creates a paragraph.

See [NewSeriesObject](#) under the Doc class for more information.

### Returns

[Pgf](#)

### Syntax

`NewSeriesPgf (previous)`

### Parameters

Parameter name	Data Type	Optional	Description
previous	Pgf	No	The object in the series after which to add the new paragraph. To add a paragraph at the start of a flow, specify the flow. To add a paragraph at the beginning of any other series, specify 0.

## GetNamedObject

### Description

Gets the object with the specified name and type. The method works with the following objects:

- AttrCondExpr
- CharFmt
- Color

- CombinedFontDefn
- Command
- CondFmt
- ElementDef
- FmtChangeList
- Menu
- MenuItemSeparator
- MasterPage
- PgfFmt
- RefPage
- RulingFmt
- TblFmt
- UnanchoredFrame (reference frame)
- VarFmt
- XRefFmt
- MarkerType

The method returns the object, on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_TypeUnNamed (-19)	Objects of the specified type do not have names.
Constants.NameNotFound (-20)	Object with the specified name and type does not exist in the specified document.

### Returns

Object

### Syntax

`GetNamedObject(objType, name)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>objType</code>	<code>int</code>	No	The type of object (for example <code>FO_TblFmt</code> ).
<code>name</code>	<code>string</code>	No	The name of the object you want to get.

## GetNamedAttrCondExpr

### Description

Gets an attribute conditional expression object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[AttrCondExpr](#)

### Syntax

GetNamedAttrCondExpr (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the AttrCondExpr object you want to get.

## GetNamedCharFmt

### Description

Gets a Character Format object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[CharFmt](#)

### Syntax

GetNamedCharFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the CharFmt object you want to get.

## GetNamedColor

### Description

Gets a Color object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[Color](#)

### Syntax

GetNamedColor (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the Color object you want to get.

## GetNamedCombinedFontDefn

### Description

Gets a Combined Font Definition object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[CombinedFontDefn](#)

### Syntax

GetNamedCombinedFontDefn (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the CombinedFontDefn object you want to get.

## GetNamedCondFmt

### Description

Gets a Conditional Format object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[CondFmt](#)

### Syntax

GetNamedCondFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the CondFmt object you want to get.

## GetNamedElementDef

### Description

Gets an Element Definition object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[ElementDef](#)

### Syntax

GetNamedElementDef (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the ElementDef object you want to get.

## GetNamedFmtChangeList

### Description

Gets a Format Change List object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[FmtChangeList](#)

### Syntax

GetNamedFmtChangeList (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the FmtChangeList object you want to get.

## GetNamedMasterPage

### Description

Gets a MasterPage object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[MasterPage](#)

### Syntax

GetNamedMasterPage (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the MasterPage object you want to get.

## GetNamedPgFmt

### Description

Gets a Paragraph Format object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[PgFmt](#)

### Syntax

GetNamedPgFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the PgFmt object you want to get.

## GetNamedRefPage

### Description

Gets a Reference Page object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[RefPage](#)

### Syntax

GetNamedRefPage (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the RefPage object you want to get.

## GetNamedRulingFmt

### Description

Gets a Ruling Format object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[RulingFmt](#)



### Syntax

GetNamedRulingFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the RulingFmt object you want to get.

## GetNamedTblFmt

### Description

Gets a Table Format object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[TblFmt](#)

### Syntax

GetNamedTblFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the TblFmt object you want to get.

## GetNamedUnanchoredFrame

### Description

Gets an Unanchored Frame object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[UnanchoredFrame](#)

### Syntax

GetNamedUnanchoredFrame (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the Unanchored Frame object you want to get.

## GetNamedVarFmt

### Description

Gets a Variable Format object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[VarFmt](#)

### Syntax

GetNamedVarFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the Variable Format object you want to get.

## GetNamedXRefFmt

### Description

Gets a Cross Reference Format object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[XRefFmt](#)

### Syntax

GetNamedXRefFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the Cross Reference Format object you want to get.

## GetNamedMarkerType

### Description

Gets a Marker Type object.

See [GetNamedObject](#) under the Doc class for more information.

### Returns

[MarkerType](#)

### Syntax

GetNamedMarkerType (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the Marker Type object you want to get.

## NewNamedObject

### Description

Creates the following named objects:

- AttrCondExpr
- CharFmt
- CombinedFontDefn
- Color
- Command
- CondFmt
- ElementDef
- FmtChangeList
- MasterPage
- Menu
- MenuItemSeparator
- Pgffmt
- RefPage
- RulingFmt
- TblFmt
- VarFmt
- XRefFmt
- MarkerType

This method uses arbitrary default properties for the objects it creates.

**Note:** When you create a new element definition, it does not appear in the Element Catalog unless you set *ElementInCatalog* to *True*. When you create a new book and specify a pathname, you must specify an absolute pathname for the name argument. To create an untitled book, pass an empty string for the name argument.

`NewNamedObject ()` returns the created named object on success.

If this method fails, the method assigns one of the following values to `FA_errno`.

**Function Summary**

FA_erno value	Meaning
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadName (-33)	Specified name for the new object is invalid.
Constants.FE_BadNew (-23)	Object cannot be created.
Constants.FE_DupName (-32)	Specified name for the new object belongs to an existing object.

**Returns**

Object of type objType

**Syntax**

NewNamedObject(objType, name)

**Parameters**

Parameter name	Data Type	Optional	Description
objType	int	No	The type of object to create (for example, Constants.FO_MasterPage or Constants.FO_PgFmt).
name	string	No	The name to be assigned to the created object.

**NewNamedAttrCondExpr****Description**

Creates a named Attribute Conditional Expression.

See [NewNamedObject](#) under the Doc class for more information.

**Returns**

[AttrCondExpr](#)

**Syntax**

NewNamedAttrCondExpr(name)

**Parameters**

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Attribute Conditional Expression.

**NewNamedCharFmt****Description**

Creates a named Character Format.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[CharFmt](#)

### Syntax

`NewNamedCharFmt (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Character Format.

## NewNamedCombinedFontDefn

### Description

Creates a named Combined Font Definition.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[CombinedFontDefn](#)

### Syntax

`NewNamedCombinedFontDefn (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Combined Font Definition.

## NewNamedColor

### Description

Creates a named Color.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[Color](#)

### Syntax

`NewNamedColor (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Color.

## NewNamedCondFmt

### Description

Creates a named Conditional Format.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[CondFmt](#)

### Syntax

NewNamedCondFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Conditional Format.

## NewNamedElementDef

### Description

Creates a named Element Definition.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[ElementDef](#)

### Syntax

NewNamedElementDef (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Element Definition.

## NewNamedFmtChangeList

### Description

Creates a named Format Change List.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[FmtChangeList](#)

### Syntax

`NewNamedFmtChangeList (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Format Change List.

## NewNamedMasterPage

### Description

Creates a named Master Page.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[MasterPage](#)

### Syntax

`NewNamedMasterPage (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Master Page.

## NewNamedPgFmt

### Description

Creates a named Paragraph Format.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[PgFmt](#)

### Syntax

`NewNamedPgFmt (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Paragraph Format.

## NewNamedRefPage

### Description

Creates a named Reference Page.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[RefPage](#)

### Syntax

`NewNamedRefPage (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Reference Page.

## NewNamedRulingFmt

### Description

Creates a named Ruling Format.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[RulingFmt](#)

### Syntax

`NewNamedRulingFmt (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Ruling Format.

## NewNamedTblFmt

### Description

Creates a named Table Format.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[TblFmt](#)



### Syntax

NewNamedTblFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Table Format.

## NewNamedVarFmt

### Description

Creates a named Variable Format.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[VarFmt](#)

### Syntax

NewNamedVarFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Variable Format.

## NewNamedXRefFmt

### Description

Creates a named Cross Reference Format.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[XRefFmt](#)

### Syntax

NewNamedXRefFmt (name)

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Cross Reference Format.

## NewNamedMarkerType

### Description

Creates a named Marker Type.

See [NewNamedObject](#) under the Doc class for more information.

### Returns

[MarkerType](#)

### Syntax

`NewNamedMarkerType (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name to be assigned to the created Marker Type.

## DeleteUnusedFmts

### Description

Deletes unused formats (character, paragraph or table) from the document.

The method returns `Constants.FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadDelete (-17)</code>	The format to be deleted is not one of character, paragraph or table.

### Returns

int

### Syntax

`DeleteUnusedFmts (objType)`

### Parameters

Parameter name	Data Type	Optional	Description
objType	int	No	The type of format to be deleted.

## DeleteUnusedCharFmt

### Description

Deletes the unused Character Formats.

**Function Summary**

See [DeleteUnusedFmts](#) under the Doc class for more information.

**Returns**

int

**Syntax**

DeleteUnusedCharFmt ()

**DeleteUnusedPgFmt****Description**

Deletes the unused Paragraph Formats.

See [DeleteUnusedFmts](#) under the Doc class for more information.

**Returns**

int

**Syntax**

DeleteUnusedPgFmt ()

**DeleteUnusedTblFmt****Description**

Deletes the unused Table Formats.

See [DeleteUnusedFmts](#) under the Doc class for more information.

**Returns**

int

**Syntax**

DeleteUnusedTblFmt ()

**Find****Description**

Performs the same actions as using the Find dialog box to search a document for text or other types of content.

The properties you can assign to `findParams` are as follows:

Property	Meaning and Possible Values
Constants.FS_FindText (1)	The text to search for.

Property	Meaning and Possible Values
Constants.FS_FindElementTag (2)	<p>Strings as follows:</p> <pre>propVal.u.ssva1.len = FV_NumFindElementItems; propVal.u.ssva1.val[FV_FindElemTag] = [an_element_tag]; propVal.u.ssva1.val[FV_FindAttrName] = [an_attribute_name]; propVal.u.ssva1.val[FV_FindAttrValue] = [an_attribute_value];</pre> <p>All of the strings must be present, but any or all may be empty.</p>
Constants.FS_FindCharFmt (3)	<p>No associated property. One or more of the following additional properties should be specified to tailor the search.</p> <ul style="list-style-type: none"> <li>• FontFamily</li> <li>• CombinedFont</li> <li>• FontSize</li> <li>• FontAngle</li> <li>• FontWeight</li> <li>• FontVariation</li> <li>• Color</li> <li>• Spread</li> <li>• Stretch</li> <li>• Language</li> <li>• Underline</li> <li>• Overline</li> <li>• Strikethrough</li> <li>• ChangeBar</li> <li>• Capitalization</li> <li>• Position</li> <li>• Tsume</li> </ul>
Constants.FS_FindPgFtag (4)	Paragraph tag.
Constants.FS_FindCharTag (5)	Character tag.
Constants.FS_FindTableTag (6)	Table tag.

Property	Meaning and Possible Values
Constants.FS_FindObject (8)	Value is one of: <ul style="list-style-type: none"> <li>• Constants.FV_FindAnyMarker (0)</li> <li>• Constants.FV_FindAnyXRef (1)</li> <li>• Constants.FV_FindUnresolvedXRef (2)</li> <li>• Constants.FV_FindAnyTextInset (3)</li> <li>• Constants.FV_FindUnresolvedTextInset (4)</li> <li>• Constants.FV_FindAnyPub (5)</li> <li>• Constants.FV_FindAnyVariable (6)</li> <li>• Constants.FV_FindAnchoredFrame (7)</li> <li>• Constants.FV_FindFootnote (8)</li> <li>• Constants.FV_FindAnyTable (9)</li> <li>• Constants.FV_FindAutomaticHyphen (10)</li> <li>• Constants.FV_FindAnyRubi (11)</li> </ul>
Constants.FS_FindMarkerOfType (9)	Marker type.
Constants.FS_FindMarkerText (10)	Marker text.
Constants.FS_FindXRefWithFormat (11)	Format string.
Constants.FS_FindNamedVariable (12)	Variable name.
Constants.FS_FindCondTextInCondTags (13)	Condition tags.
Constants.FS_FindCondTextNotInCondTags (14)	Condition tag.
Constant.FS_FindCustomizationFlags (15)	An optional parameter of type FT_Integer that may be any of the following bit flags OR'ed together: <ul style="list-style-type: none"> <li>• Constants.FF_FIND_CONSIDER_CASE (0x01)</li> <li>• Constants.FF_FIND_WHOLE_WORD (0x02)</li> <li>• Constants.FF_FIND_USE_WILDCARDS (0x04)</li> <li>• Constants.FF_FIND_BACKWARDS (0x08)</li> </ul> <p>If no customization flags are specified, the default is to search forward, to not use wildcards, to not consider case, and to not use whole words.</p>
Constants.FS_FindWrap (16)	A flag that determines whether the find operation wraps when it reaches the location where the search began. Default is True; the find operations wraps.If False, after reaching the location where the search began, the find operation returns an empty TextRange and FA_errno is set to Constants.FE_NotFound (-95).

Whenever the Find() method finds something that corresponds to a text range (a word, object anchor, marker, etc.), it returns a TextRange structure for that range. However, when searching for structure elements, you can find elements that have no corresponding text range. Structure elements for the following table parts have no corresponding text range:

- Table title
- Table head

**Function Summary**

- Table foot
- Table body
- Table row
- Table cell

When the `Find()` method finds a structure element for one of these objects, it returns an empty `TextRange` structure and sets `FA_errno` to `FE_Success`. In this case, you can get the document's `ElementSelection` property to return a corresponding `ElementRange` structure for the table part structure element.

On failure, an empty text range is returned and `FA_errno` is set to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadParameter (-43)</code>	<code>findParams</code> was empty or a parameter was improperly specified.
<code>Constants.FE_BadInsertPos (-65)</code>	The value of the <code>textLoc</code> parameter is invalid.
<code>Constants.FE_NotTextObject (-41)</code>	<code>textLoc</code> parameter is not a text location.

**Returns**

[TextRange](#)

**Syntax**

```
Find(textLoc, findParams)
```

**Parameters**

Parameter name	Data Type	Optional	Description
<code>textLoc</code>	<a href="#">TextLoc</a>	No	The text location to begin searching from.
<code>findParams</code>	<a href="#">PropVals</a>	No	A property list that specifies what to search for.

**SetElementRange****Description**

Sets an element range property.

To specify a selection that includes the root element, set `beg.parentId` to 0, `beg.childId` to the root element, and `end.childId` to 0.

The method does not return anything on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadObjId (-3)</code>	invalid object.
<code>Constants.FE_WrongProduct (-60)</code>	Current product interface is not Structured FrameMaker.

### Returns

Void

### Syntax

```
SetElementRange(propNum, setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
propNum	int	No	The property to set. Specify an ESTK defined property, such as <code>ElementSelection</code> .
setVal	<code>ElementRange</code>	No	The element range to which the property must be set.

## ElementLocToTextLoc

### Description

Returns the text location structure that corresponds with the current element location.

The method returns the text location structure containing the element location, on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_Badparameter (-43)</code>	The <code>eloc</code> parameter was empty or incorrectly specified.
<code>Constants.FE_WrongProduct (-60)</code>	Current product interface is not Structured FrameMaker.

### Returns

`TextLoc`

### Syntax

```
ElementLocToTextLoc(eloc)
```

### Parameters

Parameter name	Data Type	Optional	Description
eloc	<code>ElementLoc</code>	No	The element location structure to convert.

## UpdateVariables

### Description

Updates all the variables in a document. The method performs the same operation as clicking Update in the Variable dialog box.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	Product does not support the operation.
Constants.FE_SystemError (-42)	Could not allocate memory.

### Returns

int

### Syntax

UpdateVariables()

## TrackChangesAcceptAll

### Description

Accepts all the track changes in the specified document.

The method returns `FE_Success` on success and returns a non-zero integer on failure.

Returns `Constants.FE_AcceptRejectCalledOnWrongPreviewState (-110)` if the document is not in Preview Off state.

### Returns

int

### Syntax

TrackChangesAcceptAll()

## TrackChangesRejectAll

### Description

Rejects all the track changes in the specified document.

The method returns `FE_Success` on success and returns a non-zero integer on failure.

Returns `Constants.FE_AcceptRejectCalledOnWrongPreviewState (-110)` if the document is not in Preview Off state.

### Returns

int

### Syntax

TrackChangesRejectAll()



## UpdateXRef

### Description

Updates the cross-references in a document. It performs the same operation as clicking Update in the Cross-Reference window.

You can OR the values listed in the following tables into the `updateXRefFlags` argument.

Value	What to update
Constants.FF_XRUI_INTERNAL (0x01)	Updates only internal cross-references.
Constants.FF_XRUI_OPEN_DOCS (0x02)	Updates only cross-references whose sources are in open documents.
Constants.FF_XRUI_CLOSED_DOCS (0x04)	Updates only cross-references whose sources are in closed documents.
Constants.FF_XRUI_FORCE_UPDATE (0x08)	Updates all cross-references, regardless of whether the source document has changed.
Constants.FF_XRUI_EVERYTHING (0x01 0x02 0x04 0x08)	Updates all cross references.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	Product does not support the operation.
Constants.FE_SomeUnresolved (-42)	There are some unresolved references after the update.

### Returns

`int`

### Syntax

```
UpdateXRef(srcDoc, xref)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>srcDoc</code>	<a href="#">Doc</a>	No	The document in which to update cross-references.
<code>xref</code>	<a href="#">XRef</a>	No	Flags to indicate which cross-references to update. .

## UpdateDITAReference

### Description

Updates a DITA object

Error	Reason
Constants.FE_WrongProduct	Current FrameMaker product doesn't support the operation.

<b>Error</b>	<b>Reason</b>
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadElementId	The Element ID provided is invalid.
Constants.FE_NonDITADocument	The document provided is not a DITA document.
Constants.FE_BadParameter	The objType provided is invalid or the objType is not valid for the type of DITA document provided.
Constants.FE_UpdatedDITAREferenceFailedInvalidElementType	Update operation failed because either the element specified is not a reference type of element or it does not match the specified object type.
Constants.FE_UpdatedDITAREferenceFailedCannotResolveReference	Update operation failed because the reference cannot be resolved.
Constants.FE_UpdatedDITAREferenceFailedCannotFindReferencedFile	Update operation failed because the referenced file cannot be found at the specified location.
Constants.FE_UpdatedDITAREferenceFailedCannotOpenReferencedFile	Update operation failed because the referenced file cannot be opened from the specified location.
Constants.FE_UpdatedDITAREferenceFailedCannotConvertToFMObject	Update operation failed because the corresponding FM object cannot be created.
Constants.FE_UpdatedDITAREferenceFailed	Update operation failed.

**Returns**

**Syntax**

**Parameters**

<b>Parameter name</b>	<b>Data Type</b>	<b>Optional</b>	<b>Description</b>

**UpdateDITAResferences**

**Description**

Updates all DITA references of the specified type.

**Returns**

void

**Syntax**

UpdateDITAResferences(docId, flag);

**Parameters**

Parameter name	Data Type	Optional	Description
docId	int	No	The ID of the document containing the inset.
flag	int	No	The available flags and their values are as follows: FF_DITAUpdateAllConrefs: 0x01 FF_DITAUpdateAllXrefs: 0x02 FF_DITAUpdateAllLinks: 0x04 FF_DITAUpdateAllTopicrefs: 0x08 FF_DITAUpdateAllTopicsetrefs: 0x10 FF_DITAUpdateAllReferences: FF_DITAUpdateAllConrefs   FF_DITAUpdateAllXrefs   FF_DITAUpdateAllLinks   FF_DITAUpdateAllTopicrefs   FF_DITAUpdateAllTopicsetrefs

If UpdateDITAResources() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_WrongProduct	Current FrameMaker product doesn't support the operation.
Constants.FE_BadDocId	The Document ID provided is invalid.
Constants.FE_NonDITADocument	The Document provided is not a DITA document.

**Save**

**Description**

Saves a document.

See “[Save](#)” on page 474 under the Book class for more information.

**Returns**

Object

**Syntax**

Save(saveAsName, saveParams, saveReturnParams)

**Parameters**

Parameter name	Data Type	Optional	Description
saveAsName	string	No	The path for saving the document or book.
saveParams	PropVals	No	A property list that tells FrameMaker how to save the file and how to respond to errors and other conditions. Use GetSaveDefaultParams() or AllocatePropVals() to create and allocate memory for this property list. To use the default list, specify null.
saveReturnParams	PropVals	No	A property list that returns information about how the FrameMaker product saved the file.

## Import

### Description

Imports text or graphics into a document.

See [Import](#) under the Book class for more information.

### Returns

Object

### Syntax

```
Import(textLocP, filename, importParams, importReturnParams)
```

### Parameters

Parameter name	Data Type	Optional	Description
textLocP	<a href="#">TextLoc</a>	No	The text location at which to import the file.
filename	string	No	The full path of the file to import.
importParams	<a href="#">PropVals</a>	No	A property list specifying how to import the file and how to respond to errors and other conditions. To use the default list, specify null.
importReturnParams	<a href="#">PropVals</a>	No	A property list that provides information about how FrameMaker imported the file. This parameter must be initialized before you call Import().

## AddNewBuildExpr

### Description

Adds a Boolean conditional expression to the document.

The method returns `FE_Success`, on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_ReadOnly (-6)</code>	Document is read-only.
<code>Constants.FE_BadName (-33)</code>	Boolean conditional expression string is invalid.

### Returns

int

### Syntax

```
AddNewBuildExpr(exprName, exprCondition)
```

### Parameters

Parameter name	Data Type	Optional	Description
exprName	string	No	The name of the Boolean conditional expression.
exprCondition	string	No	The Boolean conditional expression string.

## DeleteBuildExpr

### Description

Deletes a Boolean conditional expression from the document.

The method returns `FE_Success`, on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_ReadOnly (-6)</code>	Document is read-only.
<code>Constants.FE_BadName (-33)</code>	Boolean conditional expression string is invalid.

### Returns

int

### Syntax

```
DeleteBuildExpr(exprName)
```

### Parameters

Parameter name	Data Type	Optional	Description
exprName	string	No	The name of the Boolean conditional expression to delete.

## SetActiveBuildExpr

### Description

Applies the Boolean conditional expression to the document.

The method returns `FE_Success`, on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_ReadOnly (-6)</code>	Document is read-only.
<code>Constants.FE_BadName (-33)</code>	Boolean conditional expression string is invalid.

### Returns

int

### Syntax

```
SetActiveBuildExpr (exprName)
```

### Parameters

Parameter name	Data Type	Optional	Description
exprName	string	No	The name of the Boolean conditional expression to be applied.

## GetActiveBuildExpr

### Description

Returns the name of the active expression in the document or null if none is active.

On failure, the method returns the following value to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.

### Returns

string

### Syntax

```
GetActiveBuildExpr ()
```

## GetBuildExpr

### Description

Returns the Boolean conditional expression in the document with the given name or null if none is found.

On failure, the method returns the following value to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadName ()	The parameter specified by <code>exprName</code> is invalid.

### Returns

string

### Syntax

```
GetBuildExpr (exprName)
```

### Parameters

Parameter name	Data Type	Optional	Description
exprName	string	No	The name of Boolean conditional expression.

## GetBuildExprCatalog

### Description

Returns an array of all Boolean conditional expression names in the document.

On failure, the method returns the following value to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.

### Returns

[Strings](#)

### Syntax

```
GetBuildExprCatalog()
```

## GetProps

### Description

The `GetProps()` method retrieves the properties of the Doc object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the Doc object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the Doc object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Element

### GetAttributes

#### Description

Gets an elements attributes.

The `Attributes` structure returned by `GetAttributes()` includes the attributes in the following order:

- Attributes defined in the element definition in the same order in which they are defined in the element definition.
- Undefined attributes in random order.

If an element does not have attributes, the `len` field of the `Attributes` structure is set to 0 and the `val` field is set to `NULL`.

Query the `valflags` field of the `Attributes` structure to determine whether the attribute is valid, or if there are any validation errors. The validation error flags are as follows:

Error	Reason
<code>Constants.FV_AV_REQUIRED (1)</code>	This attribute is required, but it has no value assigned to it.
<code>Constants.FV_AV_INVALID_CHOICE (2)</code>	At least one value for the attribute is not one of the allowed choices.
<code>Constants.FV_AV_INVALID_FORMAT (3)</code>	The attribute value is of the wrong type for the attribute.
<code>Constants.FV_AV_IDREF_UNRESOLVED (4)</code>	The attribute refers to an undefined ID value.
<code>Constants.FV_AV_ID_DUPLICATE_IN_DOC (5)</code>	The attribute value should be unique, but is not unique within the document.
<code>Constants.FV_AV_TOO_MANY_TOKENS (7)</code>	The attribute value has more than one token, but the attribute definition only allows one token.
<code>Constants.FV_AV_UNDEFINED (8)</code>	The attribute is not defined for the containing element.
<code>Constants.FV_AV_OUT_OF_RANGE (9)</code>	The attribute value is out of the range specified in the attribute definition.



On failure, `GetAttributes()` assigns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_WrongProduct (-60)</code>	Current product interface is not structured FrameMaker.
<code>Constants.FE_BadObjId (-3)</code>	Invalid object.

### Returns

[Attributes](#)

### Syntax

```
GetAttributes()
```

## Delete

### Description

Deletes an element.

See [Delete](#) under the `AFrame` class for more information.

### Returns

`int`

### Syntax

```
Delete()
```

## UpdateDITReferences

## GetText

### Description

Gets the text from the element.

See [GetText](#) under the `Cell` class for more information.

### Returns

[TextItems](#)

### Syntax

```
GetText(flags)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>flags</code>	<code>int</code>	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

Retrieves the properties of the element object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the element object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the element object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## ElementCatalogEntries

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

[ElementCatalogEntries](#)

#### Syntax

`concat (value)`

#### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">ElementCatalogEntries</a>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

#### Returns

[ElementCatalogEntry](#)

#### Syntax

`pop ()`

### push

#### Description

Returns the new length of the array.

#### Returns

number

#### Syntax

`push (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">ElementCatalogEntry</a>	No	The value to be pushed.

## ElementCatalogEntry

### ElementCatalogEntry

#### Description

Describes a catalog entry in an Element Catalog in FrameMaker.

#### Returns

[ElementCatalogEntry](#)

#### Syntax

```
ElementCatalogEntry(obj, flags)
```

### Parameters

Parameter name	Data Type	Optional	Description
obj	<a href="#">ElementDef</a>	No	Element definition.
flags	uint	No	Validation type.

The value of the flags field can be one of the following constants:

Flag	Meaning
Constants.FV_STRICTLY_VALID (0x01)	Catalog entry is strictly valid.
Constants.FV_LOOSELY_VALID (0x02)	Catalog entry is loosely valid.
Constants.FV_ALTERNATIVE (0x04)	Catalog entry is an alternative.
Constants.FV_INCLUSION (0x08)	Catalog entry is valid because it is an inclusion.

If no flags are set, the element is invalid at its current position.

## ElementDef

### NewTextFmtRules

#### Description

Creates a `TextFmtRules` object. The method allows you to associate the created object with a specified property of the parent object. For example, you can associate the text format rule as the text format rule of the element definition.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

### Returns

[FmtRule](#)

### Syntax

```
NewTextFmtRules ( )
```

## NewObjectFmtRules

### Description

Creates a `ObjectFmtRules` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	One

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

### Returns

[FmtRule](#)

### Syntax

```
NewObjectFmtRules()
```

## NewPrefixRules

### Description

Creates a `PrefixRules` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

### Returns

[FmtRule](#)

### Syntax

```
NewPrefixRules()
```

## NewSuffixRules

### Description

Creates a `SuffixRules` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.

Error	Reason
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

### Returns

[FmtRule](#)

### Syntax

```
NewSuffixRules()
```

## NewFirstPgfRules

### Description

Creates a `FirstPgfRules` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

### Returns

[FmtRule](#)

### Syntax

```
NewFirstPgfRules()
```

## NewLastPgfRules

### Description

Creates a `LastPgfRules` object.

The following table shows the parent object and the type of the object created by the method:

**Function Summary**

Parent	Type of Object	Number of Instances That Can be Created
ElementDef	FmtRule	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

**Returns**

`FmtRule`

**Syntax**

```
NewLastPgfrules()
```

## ElementDefIsText

**Description**

Some structural elements in FrameMaker documents are placeholders for text. For example, when a `Para` element contains text with a cross-reference element embedded in it, the ranges of text that surround the cross-reference element are treated as elements themselves. These elements are called `text nodes`.

The `ElementDefIsText()` method checks the value of an element definition to determine whether the element it is applied to is a text node.

The method returns `1` if the element definition corresponds to that of a text node, or `0` if it does not correspond to a text node.

**Returns**

`int`

**Syntax**

```
ElementDefIsText()
```

## NewElement

**Description**

Creates a structural element (`FO_Element`) in a FrameMaker document.

`NewElement()` inserts the new element at the specified location in text, using the specified element definition.



**Function Summary**

For object (noncontainer) elements, `NewElement()` inserts the appropriate type of object for the element. If there is a matching format rule, `NewElement()` uses it to format the object. Otherwise, the method uses one of the following default formats:

Object Type	Object Inserted	Format Used By Default
Constants.FV_FO_XREF (3)	Cross Reference	Undefined XRef
Constants.FV_FO_EQN (7)	Equation	Medium
Constants.FV_FO_MARKER (4)	Marker	Type 11
Constants.FV_FO_TBL (8)	Table with the format specified by the table format.	Format A if it exists; otherwise, a table with a heading row, 8 body rows, a footing row, and 5 columns.
Constants.FV_FO_SYS_VAR (2)	Variable	Filename (Long)
Constants.FV_FO_GRAPHIC (6)	A centered 1.0-inch by 1.0-inch anchored frame below the current position; cropped is off, and floating is on.	-

The method returns the created element, on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
Constants.FE_BadNew (-23)	Object cannot be created.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_NotTextObject (-41)	Object specified for the text location is not a paragraph (FO_Pgf) or a text line (FO_TextLine).
Constants.FE_OffsetNotFound (-21)	Offset specified for the text location could not be found in the specified paragraph or text line.

**Returns**

[Element](#)

**Syntax**

`NewElement(textLoc)`

**Parameters**

Parameter name	Data Type	Optional	Description
textLoc	<a href="#">TextLoc</a>	No	The text location at which to insert the new element.

**NewElementInHierarchy****Description**

Creates a structural element (FO\_Element) at a specified location in the element hierarchy of a structured FrameMaker document or book.

To create the root element for a book, you must use `NewElementInHierarchy()`. However, you cannot use `NewElementInHierarchy()` to add elements to an unstructured document. You must structure the document first by adding a root element with `NewElement()`.

**Function Summary**

For object (noncontainer) elements, `NewElementInHierarchy()` inserts the appropriate type of object for the element. If there is a matching format rule, `NewElementInHierarchy()` uses it to format the object. Otherwise, the method uses one of the following default formats:

Object Type	Object Inserted	Format Used By Default
Constants.FV_FO_XREF (3)	Cross Reference	Undefined XRef
Constants.FV_FO_EQN (7)	Equation	Medium
Constants.FV_FO_MARKER (4)	Marker	Type 11
Constants.FV_FO_TBL (8)	Table with the format and number of rows and columns specified by the table format.	Format A if it exists; otherwise, a table with a heading row, 8 body rows, a footing row, and 5 columns.
Constants.FV_FO_SYS_VAR (2)	Variable	Filename (Long)
Constants.FV_FO_GRAPHIC (6)	A centered 1.0-inch by 1.0-inch anchored frame below the current position; cropped is off, and floating is on.	-

The method returns the created element, on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
Constants.FE_BadBookId (-66)	Invalid book.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadInsertPos (-65)	<code>elemLoc</code> specifies an invalid place to insert the element; for example, it specifies a position before the highest element in the flow.

**Returns**

[Element](#)

**Syntax**

```
NewElementInHierarchy(elemLoc)
```

**Parameters**

Parameter name	Data Type	Optional	Description
<code>elemLoc</code>	<a href="#">ElementLoc</a>	No	The location at which the element is to be inserted.

**WrapElement****Description**

Inserts a structural element around the selected text and structural elements in a document.

If the flow that contains the selection is unstructured and the selection does not include the entire flow contents, `WrapElement()` wraps the flow contents into a `NoName` element before wrapping the selection into the specified element definition.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current product interface is not Structured FrameMaker.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadElementDefId(-28)	Specified element definition is invalid.
Constants.FE_BadSelectionForOperation (-59)	Current text selection is invalid for this operation.

### Returns

Void

### Syntax

WrapElement ()

## GetAttributeDefs

### Description

Gets the attribute definitions of an element definition.

On success, the method returns an `AttributeDefs` structure containing the attribute definitions.

The `attrType` field of the returned `AttributeDefs` structure identifies the attribute value's type. It can specify one of the following constants:

Constant	Value
Constant.FV_AT_STRING (0)	Any arbitrary text string.
Constant.FV_AT_STRINGS (1)	One or more arbitrary text strings.
Constants.FV_AT_CHOICES (2)	A value from a list of choices.
Constants.FV_AT_INTEGER (3)	A signed whole number (optionally restricted to a range of values).
Constants.FV_AT_INTEGERS (4)	One or more integers (optionally restricted to a range of values).
Constants.FV_AT_REAL (5)	A real number (optionally restricted to a range of values).
Constants.FV_AT_REALS (6)	One or more real numbers (optionally restricted to a range of values).
Constants.FV_AT_UNIQUE_ID (7)	A string that uniquely identifies the element.
Constants.FV_AT_UNIQUE_IDREF (8)	A reference to a UniqueID attribute.
Constants.FV_AT_UNIQUE_IDREFS (9)	One or more references to UniqueID attributes.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_WrongProduct (-60)	Current product interface is not Structured FrameMaker.
Constants.FE_BadObjId(-3)	Invalid object.

### Returns

[AttributeDefs](#)

### Syntax

```
GetAttributeDefs()
```

## Delete

### Description

Deletes an element definition.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

The `GetProps()` method retrieves the properties of an element definition.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of an element definition.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the element definition is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## ElementLoc

### ElementLoc

### Description

Contains a structure specifying an element location. The ElementLoc structure contains the returned parent element, child element and the offset within the child or parent element.

### Returns

[ElementLoc](#)

### Syntax

```
ElementLoc(parent, child, offset)
```

### Parameters

Parameter name	Data Type	Optional	Description
parent	<a href="#">Element</a>	No	Parent element.
child	<a href="#">Element</a>	No	Child element.
offset	int	No	Offset within the child or parent element.

# ElementRange

## ElementRange

### Description

A structure specifying an element's range.

### Returns

[ElementRange](#)

### Syntax

`ElementRange (beg, end)`

### Parameters

Parameter name	Data Type	Optional	Description
beg	<a href="#">ElementLoc</a>	No	Beginning of the element's range.
end	<a href="#">ElementLoc</a>	No	End of the element's range.

# Ellipse

## Delete

### Description

Deletes an ellipse.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

`Delete()`

## GetProps

### Description

Retrieves the properties of an ellipse.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

`GetProps()`

## SetProps

### Description

Sets the properties of an ellipse.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

`SetProps(setVal)`

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the ellipse is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

`ObjectValid()`

## FilterArgs

### FilterArgs

### Description

FilterArgs stores information that is used in the notification for a file-to-file filter (`FA_Note_FilterFileToFile`). The notification handler receives a pointer to this structure.

### Returns

[FilterArgs](#)

## Definition

```
FilterArgs{infile, outfile, cname, informat, outformat, binname, args}
```

## Parameters

Parameter name	Data Type	Optional	Description
infile	string	No	The file to filter.
outfile	string	No	The filter output file.
cname	string	No	The registered name of the filter.
informat	string	No	The registered incoming format.
outformat	string	No	The registered output format.
binname	string	No	The name of the filter client file.
args	Strings	No	Reserved for future use. Currently NULL.

# Flow

## Delete

### Description

Deletes a flow.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetText

### Description

Gets the text from the flow.

See [GetText](#) under the Cell class for more information.

### Returns

[TextItems](#)

### Syntax

```
GetText(flags)
```



### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

Retrieves the properties of the flow.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the flow.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the flow is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## FmtChangeList

### Delete

#### Description

Deletes a format change list.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

### Syntax

```
Delete()
```

### GetProps

#### Description

Retrieves the properties of the format change list.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

### SetProps

#### Description

Sets the properties of a format change list.

See [SetProps](#) under the AFrame class for more information.

#### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the format change list is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

`ObjectValid()`

## FmtRule

### NewFmtRuleClauses

### Description

Creates a `FmtRuleClauses` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
FmtRule	FmtRuleClause	Multiple

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

### Returns

[FmtRuleClause](#)

### Syntax

```
NewFmtRuleClauses()
```

## Delete

### Description

Deletes a format rule.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of a format rule.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of a format rule.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the format rule is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## FmtRuleClause

### NewSubFmtRule

### Description

Creates a `SubFmtRule` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
FmtRuleClause	FmtRule	One

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

### Returns

[FmtRule](#)

### Syntax

```
NewSubFmtRule()
```

## NewFmtChangeList

### Description

Creates a `FmtChangeList` object.

The following table shows the parent object and the type of the object created by the method:

Parent	Type of Object	Number of Instances That Can be Created
FmtRuleClause	FmtChangeList	One

The method returns the created object, on success. On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_BadDocId(-2)	Invalid document.
Constants.FE_BadObjId(-3)	Invalid object.
Constants.FE_BadNew(-23)	Object cannot be created.
Constants.FE_BadPropNum(-4)	The property number is invalid.
Constants.FE_WrongProduct(-60)	Current product interface is not Structured Framemaker.

### Returns

[FmtChangeList](#)

### Syntax

```
NewFmtChangeList ()
```

## Delete

### Description

Deletes a format rule clause.

See [Delete](#) under the `AFrame` class for more information.

### Returns

`int`

### Syntax

```
Delete ()
```

## GetProps

### Description

Retrieves the properties of a format rule clause.

See [GetProps](#) under the `AFrame` class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of a format rule clause.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the format rule clause is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Fn

## Delete

### Description

Deletes a footnote.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

Delete()

## GetText

### Description

Gets the text from the footnote.

See [GetText](#) under the Cell class for more information.

### Returns

[TextItems](#)

### Syntax

GetText(flags)

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

Retrieves the properties of a footnote.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

GetProps()

## SetProps

### Description

Sets the properties of a footnote.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void



### Syntax

SetProps (setVal)

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the footnote is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

ObjectValid()

## Font

### Font

### Description

Font specifies a combination of font characteristics. Each field specifies an index into a list of names in the FrameMaker product session. For example, the `family` field specifies the index of a name in the list of names specified by the session property `FontFamilyNames`.

The `weight` field specifies the index of a name in the list of names specified by the session property `FontWeightNames`.

### Returns

[Font](#)

### Syntax

Font(family, variation, weight, angle)

### Parameters

Parameter name	Data Type	Optional	Description
family	uint	No	Index of the font family.

Parameter name	Data Type	Optional	Description
variation	uint	No	Index of the font variation.
weight	uint	No	Index of the font weight.
angle	uint	No	Index of the font angle.

## Fonts

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array as the result of concatenation, with the given values to the end of the original array.

#### Returns

[Fonts](#)

#### Syntax

```
concat (value)
```

#### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Fonts</a>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

#### Returns

[Font](#)

#### Syntax

```
pop ()
```

### push

#### Description

Pushes the font on the array and the returns the new length of the array.

### Returns

number

### Syntax

push(value)

### Parameters

Parameter name	Data Type	Optional	Description
value	Font	No	TBD

## Group

A Group object represents a set of grouped objects. Group objects have the following methods.

### Delete

#### Description

Deletes a group.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

Delete()

### GetProps

#### Description

Retrieves the properties of the group.

See [GetProps](#) under the AFrame class for more information.

#### Returns

PropVals

#### Syntax

GetProps()

### SetProps

#### Description

Sets the properties of the group.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the group is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## HiddenPage

### GetProps

#### Description

Retrieves the properties of the hidden page.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

#### Syntax

```
GetProps ()
```

### SetProps

#### Description

Sets the properties of the hidden page.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the hidden page is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## ApplyPageLayout

### Description

Applies the layout of a page to another page.

PageObject is any Page object i.e., [BodyPage](#), [MasterPage](#), [HiddenPage](#).

The method returns `FE_Success` on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or fmbatch is running.
<code>Constants.FE_BadOperation (-27)</code>	Parameter specified an invalid operation.
<code>Constants.FE_BadParameter (-43)</code>	Parameter has an invalid value.
<code>Constants.FE_SystemError (-42)</code>	System Error.

### Returns

int

### Syntax

`ApplyPageLayout (srcPage)`

### Parameters

Parameter name	Data Type	Optional	Description
srcPage	PageObject	No	The page to which the layout must be applied.

## IdValuePairs

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

[Ints](#)

#### Syntax

`concat (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Ints</a>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

#### Returns

`int`

#### Syntax

`pop ()`

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

push(value)

### Parameters

Parameter name	Data Type	Optional	Description
value	int	No	TBD

## Inset

### GetIntByName

#### Description

Queries an integer facet.

`GetIntByName()` uses a transaction model to query facets. After you have finished a series of queries, you must commit the transaction by calling `GetIntByName()` to query a facet named " ".

If `GetIntByName()` returns 0, it may not indicate an error, because some facet values can be 0. To determine if a returned 0 is a property value or an error, check `FA_errno`.

Returns the specified facet value, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadObjId (-3)</code>	Invalid object.
<code>Constants.FE_BadPropNum (-4)</code>	Specified property name is invalid.
<code>Constants.FE_BadPropType (-5)</code>	Incorrect property type.
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker does not support this operation.

#### Returns

int

#### Syntax

`GetIntByName(propName)`

### Parameters

Parameter name	Data Type	Optional	Description
propName	string	No	The name of the facet to query.

## GetMetricByName

### Description

Queries a metric facet.

`GetMetricByName()` uses a transaction model to query facets. After you have finished a series of queries, you must commit the transaction by calling `GetIntByName()` to query a facet named " ".

If `GetMetricByName()` returns 0, it may not indicate an error, because some facet values can be 0. To determine if a returned 0 is a property value or an error, check `FA_errno`.

Returns the specified facet value, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadObjId (-3)</code>	Invalid object.
<code>Constants.FE_BadPropNum (-4)</code>	Specified property name is invalid.
<code>Constants.FE_BadPropType (-5)</code>	Incorrect property type.
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker does not support this operation.

### Returns

int

### Syntax

```
GetMetricByName(propName)
```

### Parameters

Parameter name	Data Type	Optional	Description
propName	string	No	The name of the facet to query.

## GetUBytesByName

### Description

Queries an unsigned bytes inset facet. The standard facets, EPSI and FrameImage, are examples of unsigned bytes facets.

If a facet contains a large amount of data, `GetUBytesByName()` only gets a portion of the data each time you call it.

To query a facet that contains a large amount of data, call `GetUBytesByName()` repeatedly until you have retrieved all the data (that is, until `UBytes.len` is 0).



**Function Summary**

`GetUBytesByName()` uses a transaction model to query facets. After you have finished a series of queries, you must commit the transaction by calling `GetIntByName()` to query a facet named " ".

On success, the method returns a `UBytes` object containing a portion of the facet's data.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadObjId (-3)</code>	Invalid object.
<code>Constants.FE_BadPropNum (-4)</code>	Specified property name is invalid.
<code>Constants.FE_BadPropType (-5)</code>	Incorrect property type.
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker does not support this operation.

**Note:** The returned `UBytes` object references memory that is allocated by the method. Use `DeallocateUBytes()` to free this memory when you are done using it.

**Returns**

`UBytes`

**Syntax**

`GetUBytesByName(propName)`

**Parameters**

Parameter name	Data Type	Optional	Description
<code>propName</code>	<code>string</code>	No	The name of the facet to query.

**SetIntByName****Description**

Sets an integer inset facet.

`SetIntByName()` uses a transaction model to set facets. After you have finished setting facets, you must commit the transaction by calling `SetIntByName()` to set a facet named " " to 0.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadObjId (-3)</code>	Invalid object.
<code>Constants.FE_BadPropNum (-4)</code>	Specified property name is invalid.
<code>Constants.FE_BadName (-33)</code>	Specified name is illegal.
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker does not support this operation.

### Returns

Void

### Syntax

SetIntByName (propName, setVal)

### Parameters

Parameter name	Data Type	Optional	Description
propName	string	No	The name of the facet to set.
setVal	int	No	The value to which the facet must be set.

## SetMetricByName

### Description

Sets a metric facet.

SetMetricByName () uses a transaction model to set facets. After you have finished setting facets, you must commit the transaction by calling SetIntByName () to set a facet named " " to 0.

The method does not return anything on success.

On failure, the method returns one of the following values to FA\_errno:

Error	Reason
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadName (-33)	Specified name is illegal.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadPropType (-5)	Incorrect property type for this function.
Constants.FE_OutOfRange (-7)	Specified property value is out of the legal range for the specified .property.
Constants.FE_ReadOnly (-6)	Property is read-only.
Constants.FE_WrongProduct (-60)	Current FrameMaker does not support this operation.

### Returns

Void

### Syntax

SetMetricByName (propName, setVal)

### Parameters

Parameter name	Data Type	Optional	Description
propName	string	No	The name of the facet to set.
setVal	int	No	The value to which the facet must be set.

## SetUBytesByName

### Description

Sets an unsigned bytes inset facet. The standard facets, EPSI and FrameImage, are examples of unsigned bytes facets.

To set a UBytes facet:

- 1 Call `SetUBytesByName()` to set the facet data.

If you are setting a facet with less than 10K of data, you need to call `SetUBytesByName()` only once.

If you are setting the facet with more than 10K of data, you should call `SetUBytesByName()` multiple times, setting a chunk of the data each time. You can size the chunks between 0 and 10K.

*Note: If you use larger chunks, the set operation goes faster. However, if the chunks are too large, you may overload your platform's interapplication communication mechanism.*

- 2 Call `SetUBytesByName()` with `propName` set to an empty string (" ").

This lets the ESTK know that you have finished setting up the facet.

The method does not return anything on success.

On failure, the method returns one of the following value to `FA_errno`:

Error	Reason
<code>Constants.FE_Transport (-1)</code>	A transportation error occurred.

### Returns

Void

### Syntax

```
SetUBytesByName(propName, setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>propName</code>	<code>string</code>	No	The name of the facet to set.
<code>setVal</code>	<code>UBytes</code>	No	The value to which the facet must be set.

## Delete

### Description

Deletes an inset.

See [Delete](#) under the AFrame class for more information.

### Returns

`int`

### Syntax

```
Delete()
```

## DeletePropByName

### Description

Deletes an inset facet. Except for inset facets, all properties are identified by integer constants. Facets are the only properties identified by name.

The method does not return anything on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_BadPropNum (-4)</code>	Specified property number is invalid.
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_BadObjId (-3)</code>	Invalid object.
<code>Constants.FE_BadPropType (-5)</code>	Incorrect property type for this function.

### Returns

Void

### Syntax

`DeletePropByName(propName)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>propName</code>	<code>string</code>	No	The name of the facet to be deleted.

## GetProps

### Description

Retrieves the properties of the inset.

See [GetProps](#) under the `AFrame` class for more information.

### Returns

[PropVals](#)

### Syntax

`GetProps()`

## SetProps

### Description

Sets the properties of the inset.

See [SetProps](#) under the `AFrame` class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the inset is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Ints

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

## concat

### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

### Returns

[Ints](#)

### Syntax

```
concat (value)
```

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Ints</a>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

int

### Syntax

```
pop()
```

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

```
push(value)
```

### Parameters

Parameter name	Data Type	Optional	Description
value	int	No	TBD

## Iterator

### GetNextIteratorItem

#### Description

Retrieves the next iterator in the document.

#### Returns

[PropVals](#)

#### Syntax

```
GetNextIteratorItem()
```

## Delete

### Description

Deletes an iterator.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the iterator.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the iterator.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the iterator is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## KeyCatalog

### NewKeyDefinition

#### Description

Adds a new key definition to the specified key catalog.

#### Returns

void

#### Syntax

```
NewKeyDefinition(key, href, srcType, srcFile, flags);
```

#### Parameters

Parameter name	Data Type	Optional	Description
key	int	No	The tag of the key for which the key definition is being added.
href	int	No	The complete path of the file that the key refers to.
srcType	int	No	The type of the file that contains the key definition. See the table below for a list of values.
srcFile	int	No	The complete path of the file that contains the key definition.
flags	int	No	Bit flags specifying information about the key definition. See the table below for a list of flags.

srcType can have one of the following values:

srcType	Meaning
Constants.FV_KeySrcTypeNone	Source file type not specified.
Constants.FV_KeySrcTypeDitamap	Source file is a DITA Map.

You can OR the following bit-flags into flags:



Bit mask	Meaning
Constants.FF_DUPLICATE_KEY_DEFINITION	The specified key definition is duplicate (that is, it already exists in the Key Catalog) and will not be used as active definition for resolving keys.
Constants.FF_FOUND_IN_REFERENCED_FILE	The specified key definition is contained in a file referenced directly or indirectly from the file that contains the key definition (srcFile).
Constants.FF_INVALID_KEY	The specified key definition is invalid due to some reason but will still be kept in the Key Catalog.

If `NewKeyDefinition()` fails, the method assigns following values to `FA_errno`:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.
Constants.FE_BadKey	The Key provided is not valid.
Constants.FE_KeyDefinitionAlreadyExists	The definition for the specified key is already available in the Key Catalog and the key definition provided is not duplicate.

## UpdateKeyDefinition

### Description

Updates the specified key definition field for the specified key in the specified key catalog.

### Returns

void

### Syntax

```
UpdateKeyDefinition(key, keyField, valuep);
```

### Parameters

Parameter name	Data Type	Optional	Description
key	int	No	The tag of the key for which the key definition is being updated.
keyField	int	No	The key field (or key information) that is being updated.
valuep	int	No	The value to update the keyField to.

The valid `keyField` values and the corresponding value type are as follows:

keyField	Value type
Constants.FV_KeydefKeyTarget	FT_String
Constants.FV_KeydefKeySrcFile	FT_String
Constants.FV_KeydefKeySrcType	FT_Integer
Constants.FV_KeydefKeyVarList	FT_Vals
Constants.FV_KeydefKeyDefaultText	FT_String
Constants.FV_KeydefKeyFoundInRefFile	FT_Integer

keyField	Value type
Constants.FV_KeydefKeyInvalid	FT_Integer
Constants.FV_KeydefKeyAttrs	FT_AttributesEx

If UpdateKeyDefinition() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.
Constants.FE_BadKey	The Key provided is not valid.
Constants.FE_KeyDefinitionDoesNotExist	The definition for the specified key is not available in the Key Catalog.
Constants.FE_BadValue	The value is either not specified or it is not as expected for the specified 'keyField'.
Constants.FE_ReadOnly	(only for keyField=FV_KeydefKeyTag or FV_KeydefKeyDuplicate). The key field cannot be changed/updated.
Constants.FE_InvAttribute	(only for keyField=FV_KeydefKeyAttrs) The Attribute information provided is not valid.
Constants.FE_WrongProduct	(only for keyField=FV_KeydefKeyAttrs) Current FrameMaker product doesn't support the operation.
Constants.FE_BadKeyField	The key field provided is not valid.

## GetKeyDefinition

### Description

Gets the specified key definition field for the specified key from the specified key catalog.

### Returns

TypedVal

### Syntax

```
GetKeyDefinition(key, keyField);
```

### Parameters

Parameter name	Data Type	Optional	Description
key	int	No	The tag of the key for which the key definition is being asked for.
keyField	int	No	The key field (or key information) that is being asked for.

The valid keyField values and the corresponding value type are as follows:

keyField	Value type
Constants.FV_KeydefKeyTag	FT_String
Constants.FV_KeydefKeyTarget	FT_String
Constants.FV_KeydefKeySrcFile	FT_String
Constants.FV_KeydefKeyDuplicate	FT_Integer

**Function Summary**

keyField	Value type
Constants.FV_KeydefKeySrcType	FT_Integer
Constants.FV_KeydefKeyVarList	FT_Vals
Constants.FV_KeydefKeyDefaultText	FT_String
Constants.FV_KeydefKeyFoundInRefFile	FT_Integer
Constants.FV_KeydefKeyInValid	FT_Integer
Constants.FV_KeydefKeyAttrs	FT_AttributesEx

If GetKeyDefinition() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.
Constants.FE_BadKey	The Key provided is not valid.
Constants.FE_KeyCatalogNotLoaded	The Key Catalog provided is currently not loaded.
Constants.FE_KeyCatalogIsStale	The Key Catalog provided is currently marked as stale and needs to be re-loaded before using.
Constants.FE_KeyDefinitionDoesNotExist	The definition for the specified key is not available in the Key Catalog.
Constants.FE_WrongProduct	(only for keyField="FV_KeydefKeyAttrs") Current FrameMaker product doesn't support the operation.
Constants.FE_BadKeyField	The key field provided is not valid.

## GetAllKeyDefinitions

### Description

Gets all the key definitions from the specified key catalog.

### Returns

TypedVals

### Syntax

```
GetAllKeyDefinitions (filterType) ;
```

### Parameters

Parameter name	Data Type	Optional	Description
filterType	int	No	Specifies the kind of key fields to get for each key definition. filterType can have the following values:  FV_KeyDefFieldsTypePrimary: Get only the primary key fields (Tag, Target, SrcFile, and Duplicate)  FV_KeyDefFieldsTypeAll: Get all key fields.

Returns the information in a TypedVals structure as follows: FieldTag is of type FT\_Integer. FieldValue is of type as specified in the table below.

**Function Summary**

FieldTag value	FieldValue type
Constants.FV_KeydefKeyAttrs	FT_AttributesEx
Constants.FV_KeydefKeyDefaultText	FT_String
Constants.FV_KeydefKeyDuplicate	FT_Integer
Constants.FV_KeydefKeyFoundInRefFile	FT_Integer
Constants.FV_KeydefKeyInValid	FT_Integer
Constants.FV_KeydefKeySrcFile	FT_String
Constants.FV_KeydefKeySrcType	FT_Integer
Constants.FV_KeydefKeyTag	FT_String
Constants.FV_KeydefKeyTarget	FT_String
Constants.FV_KeydefKeyVarList	FT_Vals

If GetAllKeyDefinitions() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.
Constants.FE_KeyCatalogNotLoaded	The Key Catalog provided is currently not loaded.
Constants.FE_KeyCatalogIsStale	The Key Catalog provided is currently marked as stale and needs to be re-loaded before using.
Constants.FE_BadFilterType	The filter type provided is not valid.

## DeleteAllKeyDefinitions

### Description

Deletes all the key definitions in the specified key catalog.

### Returns

void

### Syntax

```
DeleteAllKeyDefinitions();
```

If DeleteAllKeyDefinitions() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.

## GetAllKeys

### Description

Gets all the key tags from the specified key catalog.

**Function Summary****Returns**

Strings

**Syntax**

```
GetAllKeys ();
```

If GetAllKeys() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadObjId	The ID provided does not specify a Key Catalog.
Constants.FE_KeyCatalogNotLoaded	The Key Catalog provided is currently not loaded.
Constants.FE_KeyCatalogIsStale	The Key Catalog provided is currently marked as stale and needs to be re-loaded before using.

**Delete****Description**

Deletes a key catalog.

See [Delete](#) under the AFrame class for more information.

**Returns**

int

**Syntax**

```
Delete ();
```

**GetProps****Description**

Retrieves the properties of the key catalog.

See [GetProps](#) under the AFrame class for more information.

**Returns**

[PropVals](#)

**Syntax**

```
GetProps ();
```

**SetProps****Description**

Sets the properties of the key catalog.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the key catalog is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Line

### Delete

### Description

Deletes a line.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the line.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the line.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the line is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Marker

### Delete

### Description

Deletes a marker.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

Delete()

## GetProps

### Description

Retrieves the properties of the marker.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

GetProps()

## SetProps

### Description

Sets the properties of the marker.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

SetProps(setVal)

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The properties list.

## ObjectValid

### Description

Returns 1 if the marker is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int



### Syntax

`ObjectValid()`

## MarkerType

### Delete

#### Description

Deletes a marker type.

See [Delete](#) under the AFrame class for more information.

#### Returns

`int`

### Syntax

`Delete()`

### GetProps

#### Description

Retrieves the properties of the marker type.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

### Syntax

`GetProps()`

### SetProps

#### Description

Sets the properties of the marker type.

See [SetProps](#) under the AFrame class for more information.

#### Returns

`Void`

### Syntax

`SetProps(setVal)`

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the marker type is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## MasterPage

### Delete

### Description

Deletes a master page.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

### GetProps

### Description

Retrieves the properties of the master page.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the master page.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the master page is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## ApplyPageLayout

### Description

Copies the layout of one master page to another.

PageObject is any Page object i.e., [BodyPage](#), [MasterPage](#), [HiddenPage](#).

See [BodyPage](#) under the Body Page class for more information.

### Returns

int

### Syntax

```
ApplyPageLayout (srcPage)
```

### Parameters

Parameter name	Data Type	Optional	Description
srcPage	PageObject	No	The page to which the layout is to be applied.

## Math

### Delete

#### Description

Deletes a math object.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

### GetProps

#### Description

Retrieves the properties of the math object.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

#### Syntax

```
GetProps()
```

### SetProps

#### Description

Sets the properties of the math object.

See [SetProps](#) under the AFrame class for more information.

#### Returns

Void

#### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the math object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## MathML

### Delete

### Description

Deletes a math object.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

### GetProps

### Description

Retrieves the properties of the math object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the math object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the math object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Menu

### DefineAndAddCommand

#### Description

Defines a command (FO\_Command object) and adds it to a menu or menu bar.

The following table lists some FrameMaker product menus and the names you use to specify them:

Menu Title	Menu Name
Edit	EditMenu
File	FileMenu
Format	FormatMenu

Menu Title	Menu Name
Graphics	GraphicsMenu
Special	SpecialMenu
Table	TableMenu
View	ViewMenu
Help	!HelpMenu

If you call `DefineAndAddCommand()` and specify the name of a command that is already defined in the user's menu configuration files, FrameMaker gives precedence to the definition in the configuration files.

If the configuration files assign a label or a shortcut to the command, then FrameMaker uses it instead of the one you specify. If the command is already a menu item, FrameMaker ignores the menu that you specify and leaves the menu item where it is.

**Note:** *If you want to add a command to more than one menu, do not call `DefineAndAddCommand()` repeatedly to add the command to the menus. To add a command to multiple menus, define the command first by calling `DefineCommand()`—or call `iDefineAndAddCommand()`, if you want to define and add the command to a menu at the same time—and then call `AddCommandToMenu()` to add the command to other menus.*

The method returns the command, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specified an invalid operation.
<code>Constants.FE_NotMenu (-72)</code>	The specified menu does not exist.
<code>Constants.FE_BadParameter (-43)</code>	The specified parameter has an invalid value.
<code>Constants.FE_SystemError (-42)</code>	System Error

### Returns

[Command](#)

### Syntax

```
DefineAndAddCommand(cmdNum, name, label, shortcut)
```

**Parameters**

Parameter name	Data Type	Optional	Description
cmdNum	number	No	The integer that the FrameMaker product passes to your client's <code>Command()</code> function when the user chooses the menu item or types the keyboard shortcut for the command.  The value must be unique for each command in your client, but need not be unique for different clients.
name	string	No	A unique name for the command. If the user or a client has already defined a command or menu with this name, the new command replaces it.
label	string	No	The title of the command as it appears on the menu.
shortcut	string	No	The keyboard shortcut sequence. Many FrameMaker product commands use shortcuts beginning with Escape (\!). To specify Escape when you create a command, use \\! in the string you pass to shortcut.

**AddCommandToMenu**

**Description**

Adds a FrameMaker product command or a client-defined command to a menu.

`AddCommandToMenu()` adds the command at the bottom of the specified menu. To change a command's position on a menu, set its `PrevMenuItemInMenu` and `NextMenuItemInMenu` properties.

To add a command that you have created, set the `command` parameter to the object returned by the `DefineCommand()` call that created the command. To add a FrameMaker product command, you must get its object. To get its object, call `GetNamedObject()` with the `objectName` parameter set to its name.

The method returns the command, on success.

On failure, the method returns one of the following values to `FA_erno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specified an invalid operation.
<code>Constants.FE_NotCommand (-73)</code>	The <code>command</code> parameter did not specify a command.
<code>Constants.FE_NotMenu (-72)</code>	The specified menu does not exist.
<code>Constants.FE_BadParameter (-43)</code>	The specified parameter has an invalid value.
<code>Constants.FE_SystemError (-42)</code>	System Error

**Returns**

number

**Syntax**

`AddCommandToMenu (command)`



**Parameters**

Parameter name	Data Type	Optional	Description
command	Command	No	The command to add.

**AddMenuToMenu**

**Description**

Adds a FrameMaker product menu or a menu that you have created to another menu or menu bar.

To add a menu to one of FrameMaker menus or menu bars, you must get the menu or menu bar's object. To get its object, call `GetNamedObject()` with the `objectName` parameter set to its name.

*Note:* Your menu appears only on the menu bar you specify. For example, if you only add a menu to the `!MakerMainMenu` menu bar, the menu will not appear if the user switches to quick menus. For your menu to appear after the user has switched to quick menus, you must also add it to `!QuickMakerMainMenu`.

The following table lists the types of menus you can add a menu to and how FrameMaker implements the added menu.

Type of menu or menu bar you are adding a menu to	How FrameMaker implements the added menu	FrameMaker product adds the menu
Menu bar	Pull-down menu	At the right of the menu bar.
Pull-down menu	Pull-right menu	At the bottom of the pull-down menu.
Pop-up menu	Pull-right menu	At the bottom of the pop-up menu.
Pull-right menu	Pull-right menu	At the bottom of the pull-right menu.

To change a menu's position on a menu or menu bar after you add it, set its `PrevMenuItemInMenu` and `NextMenuItemInMenu` properties.

The method returns the command, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_NotMenu (-72)</code>	The specified menu does not exist.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specified an invalid operation.
<code>Constants.FE_BadParameter (-43)</code>	The specified parameter has an invalid value.
<code>Constants.FE_SystemError (-42)</code>	System Error

**Returns**

int

**Syntax**

`AddMenuToMenu (menu)`

**Parameters**

Parameter name	Data Type	Optional	Description
menu	Menu	No	The new menu to be added.

**DefineAndAddMenu**

**Description**

Defines a menu (FO\_Menu object) and adds it to another menu.

To add a menu to one of FrameMaker’s menus or menu bars, you must get the menu or menu bar’s object. To get its object, call `GetNamedObject ()` with the `objectName` parameter set to its name.

The following table lists some of the menu bars that you can add menus to and the strings that specify them. Menu bar names preceded by an exclamation mark (!) cannot be removed by the user.

FrameMaker menubar	Menubar string
Menu bar for documents (complete menus)	!MakerMainMenu
Menu bar for documents (quick menus)	!QuickMakerMainMenu
Menu bar for documents (custom menus)	!CustomMakerMainMenu
Menu bar for books (complete menus)	!BookMainMenu
Menu bar for books (quick menus)	!QuickBookMainMenu
Structure menu bar (structured product interface only)	!StructureViewMainMenu
Pop-up menu for documents (complete menus; UNIX only)	!MakerPopup
Pop-up menu for documents (quick menus; UNIX only)	!QuickMakerPopup
View-only menu bar	!ViewOnlyMainMenu
View-only pop-up menu (UNIX only)	!ViewerPopup

**Note:** Your menu appears only on the menu bar you specify. For example, if you only add a menu to the `!MakerMainMenu` menu bar, the menu will not appear if the user switches to quick menus. For your menu to appear after the user has switched to quick menus, you must also add it to `!QuickMakerMainMenu`.

If you call `DefineAndAddMenu ()` and specify the name of a menu that is already defined in the user’s menu configuration files, FrameMaker gives precedence to the definition in the configuration files.

If the configuration files assign a label to the menu, FrameMaker uses it instead of the one you specify. If the menu is already on a menu or menu bar, FrameMaker ignores the menu that you specify and leaves the menu where it is.

The following table lists the types of menus you can add a menu to and how FrameMaker implements the added menu.

Type of menu or menu bar you are adding a menu to	How FrameMaker implements the added menu	FrameMaker product adds the menu
Menu bar	Pull-down menu	At the right of the menu bar.
Pull-down menu	Pull-right menu	At the bottom of the pull-down menu.
Pop-up menu	Pull-right menu	At the bottom of the pop-up menu.
Pull-right menu	Pull-right menu	At the bottom of the pull-right menu.

The method returns the menu, on success.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_NotMenu (-72)</code>	The specified menu does not exist.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specified an invalid operation.
<code>Constants.FE_BadParameter (-43)</code>	The specified parameter has an invalid value.
<code>Constants.FE_SystemError (-42)</code>	System Error

## Returns

[Menu](#)

## Syntax

```
DefineAndAddMenu(name, label)
```

## Parameters

Parameter name	Data Type	Optional	Description
<code>name</code>	<code>string</code>	No	A unique name for the new menu. If the user or an ESTK client has already defined a command or menu with this name, the new menu replaces it.
<code>label</code>	<code>string</code>	No	The title of the menu as it appears on the menu bar or menu.

## MenuItemInMenu

### Description

Determines if a menu item or menu is on a menu or menu bar.

On success, the method returns the menu on which the menu item is found.

On failure, the method returns one of the following values to `FA_errno`:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specified an invalid operation.
<code>Constants.FE_BadParameter (-43)</code>	The specified parameter has an invalid value.

## Returns

[Menu](#)

## Syntax

```
MenuItemInMenu(menuitem, recursive)
```

### Parameters

Parameter name	Data Type	Optional	Description
menuItem	MenuItemComponent	No	The menu item or menu to find.
recursive	bool	No	Specifies whether to search the submenus of the menu or menu bar specified by <code>menuItem</code> . Specify <code>True</code> to search them.

## Delete

### Description

Deletes a menu or menu item.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the menu or menu item.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the menu or menu item.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the menu or menu item is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Delete

### Description

Deletes a menu or menu item.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## MenuItemSeparator

### GetProps

### Description

Retrieves the properties of the menu item separator.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the menu item separator.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the menu item separator is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Delete

### Description

Deletes a menu item separator.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## Metrics

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

[Metrics](#)

#### Syntax

`concat (value)`

#### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Metrics</a>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

#### Returns

int

#### Syntax

`pop ()`

### push

#### Description

Returns the new length of the array.

#### Returns

number

#### Syntax

`push (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	int	No	The value to push into the array.

## Pgf

### Delete

#### Description

Deletes a paragraph.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

Delete()

### GetText

#### Description

Gets the text from the paragraph.

See [GetText](#) under the Cell class for more information.

#### Returns

[TextItems](#)

#### Syntax

GetText(flags)

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

### GetProps

#### Description

Retrieves the properties of the paragraph.

See [GetProps](#) under the AFrame class for more information.



### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the paragraph.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the paragraph is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## PgfFmt

### Delete

### Description

Deletes a paragraph format.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

Delete()

## GetProps

### Description

Retrieves the properties of the paragraph format.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

GetProps()

## SetProps

### Description

Sets the properties of the paragraph format.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

SetProps(setVal)

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the paragraph format is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

`ObjectValid()`

## Point

### Point

#### Description

Point describes an individual coordinate pair. FrameMaker measures coordinates from the upper-left corner of the parent frame

#### Returns

[Point](#)

#### Syntax

`Point(x, y)`

#### Parameters

Parameter name	Data Type	Optional	Description
x	int	No	X coordinate
y	int	No	Y coordinate

## Points

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

[Points](#)

#### Syntax

`concat(value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Points</a>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

[Point](#)

### Syntax

```
pop()
```

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

```
push(value)
```

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Point</a>	No	The value to push into the array.

## Polygon

### Delete

#### Description

Deletes a polygon.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

### Syntax

Delete()

## GetProps

### Description

Retrieves the properties of the polygon.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

GetProps()

## SetProps

### Description

Sets the properties of the polygon.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

SetProps(setVal)

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the polygon is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

ObjectValid()

# Polyline

## Delete

### Description

Deletes a polyline.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the polyline.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the polyline.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the polyline is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## PropIdent

### PropIdent

### Description

PropIdent provides a property identifier. Properties can be identified by either a name or a number (integer constant). ESTK provides defined constants for property numbers (for example, Fill and Height). Only inset properties (facets) are identified by names. If a property is identified by a name, `PropIdent.num` is set to 0. If a property is identified by a number, `PropIdent.name` is set to a null string.

### Returns

[PropIdent](#)

### Syntax

```
PropIdent(num, name)
```

### Parameters

Parameter name	Data Type	Optional	Description
num	int	No	Property number.
name	string	No	Property name.

## PropVal

### PropVal

### Description

Describes a property-value pair.

### Returns

[PropVal](#)

### Syntax

```
PropVal(propIdent, propVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
propIdent	<a href="#">PropIdent</a>	No	Property identifier.
propVal	<a href="#">TypedVal</a>	No	Property value.

## PropVals

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

[PropVals](#)

#### Syntax

```
concat(value)
```

#### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">PropVals</a>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

#### Returns

[PropVal](#)

#### Syntax

```
pop()
```



## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

`push(value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">PropVal</a>	No	The value to push into the array.

## Rectangle

### Delete

#### Description

Deletes a rectangle.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

`Delete()`

### GetProps

#### Description

Retrieves the properties of the rectangle.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

#### Syntax

`GetProps()`

## SetProps

### Description

Sets the properties of the rectangle.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the rectangle is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## RefPage

### Delete

#### Description

Deletes a reference page.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the reference page.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the reference page.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the reference page is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## ApplyPageLayout

### Description

PageObject is any Page object i.e., [BodyPage](#), [MasterPage](#), [HiddenPage](#).

See [BodyPage](#) under the BodyPage class for more information.

### Returns

int

### Syntax

```
ApplyPageLayout (srcPage)
```

### Parameters

Parameter name	Data Type	Optional	Description
srcPage	PageObject	No	The page to which the layout is to be applied.

## RoundRect

### Delete

#### Description

Deletes a round rectangle.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

### GetProps

#### Description

Retrieves the properties of the round rectangle.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

#### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the rounded rectangle.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the rounded rectangle is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## Row

### AddRows

#### Description

Adds one or more rows to a table.

The following table lists the constants you can specify for the `direction` parameter:

Direction	Meaning
Constants.FV_Above (1240)	Add rows above the current reference row.
Constants.FV_Below (1241)	Add rows below the current reference row
Constants.FV_Body (1237)	Add rows at the bottom of the existing body rows.

**Function Summary**

Direction	Meaning
Constants.FV_Footing (1239)	Add rows at the bottom of the existing footer rows.
Constants.FV_Heading (1238)	Add rows at the bottom of the existing heading rows

The method returns `FE_Success` on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker product does not support this operation.
Constants.FE_BadDocId (-2)	Invalid document.
Constants.FE_BadObjId (-3)	Invalid row number.
Constants.FE_BadOperation (-27)	Parameters specified an invalid operation.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.

**Returns**

`int`

**Syntax**

```
AddRows(direction, numNewRows)
```

**Parameters**

Parameter name	Data Type	Optional	Description
<code>direction</code>	<code>int</code>	No	The direction from the reference row in which to add rows.
<code>numNewRows</code>	<code>int</code>	No	The number of rows to add.

**Delete****Description**

Deletes a row.

See [Delete](#) under the `AFrame` class for more information.

**Returns**

`int`

**Syntax**

```
Delete()
```

**GetProps****Description**

Retrieves the properties of the row.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps ()
```

## SetProps

### Description

Sets the properties of the row.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps (setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the row is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid ()
```

## Rubi

### Delete

### Description

Deletes a rubi object.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

Delete ()

## GetText2

### Description

Gets the text from all the objects available to `GetText ()`, as well as an extended set of document objects that includes the rubi object.

`TextItems.dataType` can be one of the following constants (in addition to the constants returned by `GetText ()`):

Value	Representation	Data
Constants.FTI2_RubiTextBegin (0x80000001)	The beginning of rubi text.	The FO_Rubi object for the rubi composite that contains the rubi text.
Constants.FTI2_RubiTextEnd (0x80000002)	The end of rubi text.	The FO_Rubi object for the rubi composite that contains the rubi text.
Constants.FTI2_RubiCompositeBegin (0x80000004)	The beginning of a rubi composite.	FO_Rubi object.
Constants.FTI2_RubiCompositeEnd (0x80000008)	The end of a rubi composite.	FO_Rubi object.

See [GetText](#) under the Cell class for more information.

### Returns

[TextItems](#)

### Syntax

GetText2 (flags, flags2)

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.
flags2	int	No	Bit flags for the extended set of text item types that specify the type of text items to retrieve. To get specific types of text items from the extended set, OR the constants that represent them (for example, use a bitwise OR to combine <code>Constants.FTI2_RubiTextBegin (0x80000001)</code> and <code>Constants.FTI2_RubiTextEnd (0x80000002)</code> ) into <code>flags2</code> . To get all types of the extended set of text items, specify -1.



## GetProps

### Description

Retrieves the properties of the rubi object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the rubi object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the rubi object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

# RulingFmt

## Delete

### Description

Deletes a ruling format.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

Retrieves the properties of the ruling format.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the ruling format.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the ruling format is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

ObjectValid()

## Strings

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to concat(), each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

[Strings](#)

#### Syntax

concat (value)

#### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Strings</a>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

string

### Syntax

pop ()

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

`push(value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	string	No	The value to push into the array.

## SubCol

### Delete

#### Description

Deletes a subcol object.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

`Delete()`

### GetText

#### Description

Gets the text from the subcol object.

See [GetText](#) under the Cell class for more information.

#### Returns

[TextItems](#)

#### Syntax

`GetText(flags)`

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

Retrieves the properties of the subcol object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

Sets the properties of the subcol object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the subcol object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

`ObjectValid()`

## Tab

### Tab

#### Description

Describes an individual tab. Note that the character specified by `decimal` must be a single byte character.

The `type` field can contain one of the following constants:

Constant	Tab Type
<code>Constants.FV_TAB_LEFT (0x1)</code>	Left tab.
<code>Constants.FV_TAB_CENTER (0x2)</code>	Center tab.
<code>Constants.FV_TAB_RIGHT (0x3)</code>	Right tab.
<code>Constants.FV_TAB_DECIMAL (0x4)</code>	Decimal tab.
<code>Constants.FV_TAB_RELATIVE_LEFT (0x5)</code>	Relative left tab (allowed only for format change lists).
<code>Constants.FV_TAB_RELATIVE_CENTER (0x6)</code>	Relative center tab (allowed only for format change lists).
<code>Constants.FV_TAB_RELATIVE_RIGHT (0x7)</code>	Relative right tab (allowed only for format change lists).
<code>Constants.FV_TAB_RELATIVE_DECIMAL (0x8)</code>	Relative decimal tab (allowed only for format change lists).

#### Returns

[Tab](#)

#### Syntax

`Tab(x, type, leader, decimal)`

#### Parameters

Parameter name	Data Type	Optional	Description
<code>x</code>	<code>int</code>	No	Offset from the left margin.
<code>type</code>	<code>uint</code>	No	Type of tab.
<code>leader</code>	<code>string</code>	No	String that appears before the tab.
<code>decimal</code>	<code>uint</code>	No	Character to align tab around - For example " , ".

## Tabs

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

## concat

### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

### Returns

[Tabs](#)

### Syntax

`concat (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Tabs</a>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

[Tab](#)

### Syntax

`pop ()`

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

`push (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">Tab</a>	No	

# Tbl

## AddCols

### Description

Adds columns to a table.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	invalid document.
Constants.FE_BadParameter (-43)	A parameter specified is invalid.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadOperation (-27)	The method specified an invalid operation.

### Returns

`int`

### Syntax

`AddCols(refColNum, direction, numNewCols)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>refColNum</code>	<code>int</code>	No	The column at which to start adding columns. The columns are numbered from left to right starting with column 0.
<code>direction</code>	<code>int</code>	No	The direction from the reference column in which to add columns. To add columns to the left of the reference column, specify <code>FV_Left</code> . To add them to the right, specify <code>FV_Right</code> .
<code>numNewCols</code>	<code>int</code>	No	The number of columns to add.

## DeleteCols

### Description

Deletes columns from a table. To delete an entire table, use `Delete()`.

The method deletes the column specified by `delColNum` and `(numDelCols-1)` columns to the right of it.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the requested operation.
Constants.FE_BadOperation (-27)	The method specified an invalid operation.



**Function Summary**

Error	Reason
Constants.FE_BadDocId (-2)	invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_TableInLockedTi (-101)	The table is in a locked text inset.

**Returns**

int

**Syntax**

DeleteCols(delColNum, numDelCols)

**Parameters**

Parameter name	Data Type	Optional	Description
delColNum	int	No	The first column to delete. Columns are numbered from left to right, starting with 0.
numDelCols	int	No	The number of columns to delete.

**DeleteRows****Description**

Deletes rows from a table. Like the Delete command in the FrameMaker product user interface, DeleteRows() does not allow you to delete more than one type of row at time.

The range of rows you specify must be all body rows, all header rows, or all footer rows.

The method deletes `refRow` and (`numDelRows` - 1) rows below it.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker version does not support the requested operation.
Constants.FE_BadOperation (-27)	The method specified an invalid operation.
Constants.FE_BadDocId (-2)	invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_OutOfRange (-7)	The <code>refRow</code> parameter does not specify a row in the table, or the specified range includes more than one type of row (for example, header rows and body rows).

**Returns**

int

**Syntax**

DeleteRows(refRow, numDelRows)

**Parameters**

Parameter name	Data Type	Optional	Description
refRow	Row	No	The first row to delete.
numDelRows	int	No	The number of rows to delete, including the value passed in refRow.

**MakeTblSelection**

**Description**

Selects a range of cells in a table.

To select an entire table, including the table title, set the topRow parameter to Constants.FF\_SELECT\_WHOLE\_TABLE (0x80000000). MakeTblSelection() ignores the values for the other parameters.

***Note:** MakeTblSelection() cannot select different types of rows at the same time, unless you set topRow to Constants.FF\_SELECT\_WHOLE\_TABLE (0x80000000) or you set topRow and bottomRow to select one or more entire columns.*

The method returns FE\_Success on success.

On failure, the method sets FA\_errno to one of the following values:

Error	Reason
Constants.FE_BadDocId (-2)	invalid document.
Constants.FE_BadObjId (-3)	Invalid object.
Constants.FE_BadOperation (-27)	The method specified an invalid operation.
Constants.FE_OutOfRange (-7)	The specified row or column is invalid.
Constants.FE_BadParameter (-43)	The parameter specified is invalid.

**Returns**

int

**Syntax**

MakeTblSelection(topRow, bottomRow, leftCol, rightCol)

**Parameters**

Parameter name	Data Type	Optional	Description
topRow	int	No	The number of the first row in the selection. The rows are numbered from top to bottom, starting with 0 (including heading rows). To select the entire table, specify Constants.FF_SELECT_WHOLE_TABLE (0x80000000).
bottomRow	int	No	The number of the last row in the selection.
leftCol	int	No	The number of the leftmost column in the selection. The columns are numbered from left to right, starting with 0.
rightCol	int	No	The number of the rightmost column in the selection.

## Delete

### Description

Deletes the specified table.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
Delete()
```

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified table.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the specified table.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the table is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

```
ObjectValid()
```

## TblFmt

The method uses an `FO_TblFmt` object to represent each table format in a document.

### Delete

Deletes the specified table format.

See “[Delete](#)” on page 453 under the AFrame class.

### GetProps

See “[GetProps](#)” on page 453 under the AFrame class.

### SetProps

See “[SetProps](#)” on page 454 under the AFrame class.

### ObjectValid

See “[ObjectValid](#)” on page 455 under the AFrame class.

## TextFrame

### Delete

#### Description

Deletes the specified text frame.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

### Syntax

Delete ()

## GetText

### Description

Gets the text from the text frame.

See [GetText](#) under the Cell class for more information.

### Returns

[TextItems](#)

### Syntax

GetText (flags)

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

The `GetProps ()` method retrieves the properties of the specified text frame.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

GetProps ()

## SetProps

### Description

The `SetProps ()` method sets the properties of the specified text frame.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

SetProps (setVal)

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the text frame is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

`ObjectValid()`

## TextItems

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

[TextItems](#)

#### Syntax

`concat (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">TextItems</a>	No	Can also be arrays.

### pop

#### Description

Removes the last element from the array.

### Returns

[TextItem](#)

### Syntax

`pop()`

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

`push(value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<a href="#">TextItem</a>	No	The value to push into the array.

## TextLine

### Delete

#### Description

Deletes the specified text line.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

`Delete()`

### GetText

#### Description

Gets the text from the text line.

See [GetText](#) under the Cell class for more information.

### Returns

[TextItems](#)

### Syntax

`GetText (flags)`

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified text line.

See [GetProps](#) under the `AFrame` class for more information.

### Returns

[PropVals](#)

### Syntax

`GetProps()`

## SetProps

### Description

The `SetProps()` method sets the properties of the specified text line.

See [SetProps](#) under the `AFrame` class for more information.

### Returns

Void

### Syntax

`SetProps (setVal)`

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the text line is valid, else returns 0.



See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

ObjectValid()

## TextLoc

### TextLoc

#### Description

TextLoc specifies a location within the text of a paragraph or a graphic text line.

#### Returns

[TextLoc](#)

#### Syntax

TextLoc(obj, offset)

#### Parameters

Parameter name	Data Type	Optional	Description
obj	any	No	FO_Pgf or FO_TextLine object.
offset	int	No	Characters from the beginning of the object.

## TextRange

### TextRange

#### Description

TextRange specifies a text range. A text range can span paragraphs. However, it cannot span graphic text lines or flows.

*Note:* `beg.offset` and `end.offset` fields of a `TextRange` object can specify offsets relative to the beginning and end of an object. In addition, they can use the special value `Constants.FV_OBJ_END_OFFSET (0x50000000)`.

*Note:* `Constants.FV_OBJ_END_OFFSET (0x50000000)` specifies the offset of the last character in the object containing the text range.

#### Returns

[TextRange](#)

### Syntax

TextRange (beg, end)

### Parameters

Parameter name	Data Type	Optional	Description
beg	<a href="#">TextLoc</a>	No	The beginning of the range.
end	<a href="#">TextLoc</a>	No	The end of the range.

## TiApiClient

An `FO_TiApiClient` object represents text imported by an FDK client.

### GetText

See “[GetText](#)” on page 493 under the Cell class.

### GetProps

See “[GetProps](#)” on page 453 under the AFrame class.

### SetProps

See “[SetProps](#)” on page 454 under the AFrame class.

### ObjectValid

See “[ObjectValid](#)” on page 455 under the AFrame class.

## DeleteTextInsetContents

### Description

This method deletes the text in a text inset. You must unlock a text inset before you call this function to delete its contents. After you are done, you must relock the text inset.

Call this method on the text inset for which you want to delete the contents.

If this method fails, the method assigns one of the following values to `FA_errno`.

Error	Meaning
Constants.FE_BadDelete (-17)	Specified text couldn't be deleted
Constants.FE_BadDocId (-2)	Invalid document ID
Constants.FE_BadObjId (-3)	Invalid object ID
Constants.FE_BadSelectionForOperation (-59)	The specified text inset is locked

### Returns

int

### Syntax

```
DeleteTextInsetContents ()
```

## UpdateTextInset

### Description

Updates the contents of a stale text inset. It determines whether an inset is stale by comparing the inset's `LastUpdate` property with the modification date of the inset's source file. `UpdateTextInset ()` does not update a text inset unless it is stale. To make a text inset stale, set its `LastUpdate` property to **0**. `UpdateTextInset ()` does not update graphic insets (`FO_Inset` objects).

Call this method on the text inset to be updated.

This method returns `FE_Success` if it succeeds or `FE_SomeUnresolved` if some text insets were unresolved. If `UpdateTextInset ()` fails, it assigns one of the following values to `FA_errno`.

Error	Meaning
<code>Constants.FE_BadDocId (-2)</code>	Invalid document ID
<code>Constants.FE_BadFileType (-83)</code>	The inset specifies a file that does not match the import type (for example, the inset imports a binary document but the file is a text file or doesn't exist)
<code>Constants.FE_SomeUnresolved (-22)</code>	Some text insets were unresolved
<code>Constants.FE_WrongProduct (-60)</code>	Product doesn't support the specified operation
<code>Constants.FE_SystemError (-42)</code>	Couldn't allocate memory

### Returns

int

### Syntax

```
UpdateTextInset ()
```

## ConvertToText

### Description

This method is called upon a text inset (Examples: `TiApiClient ()`, `TiFlow ()`, `TiText ()`, `TiTextTable ()`) object and converts it to a text. This method deletes the text inset and removes the locked text range around it without modifying the content.

### Returns

Void

### Syntax

```
ConvertToText ()
```

**Function Summary**

## Delete

See “Delete” on page 453 under the AFrame class.

## DeletePropByName

### Description

Deletes an inset facet. Except for inset facets, all properties are identified by integer constants. Facets are the only properties identified by name.

If `DeletePropByName()` fails, the method assigns one of the following values to `FA_erno`:

Error	Meaning
<code>Constants.FE_BadPropNum (-4)</code>	Specified property number is invalid
<code>Constants.FE_BadDocId (-2)</code>	Invalid document ID
<code>Constants.FE_BadObjId (-3)</code>	Invalid object ID
<code>Constants.FE_BadPropType (-5)</code>	Incorrect property type for this function

### Returns

Void

### Syntax

```
DeletePropByName(propName)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>propName</code>	<code>string</code>	No	The name of the property to delete

## TiFlow

An `FO_TiFlow` object represents text imported from a FrameMaker document or a MIF file.

## GetText

See “GetText” on page 493 under the Cell class.

## GetProps

See “GetProps” on page 453 under the AFrame class.

## SetProps

See “SetProps” on page 454 under the AFrame class.

## ObjectValid

See “[ObjectValid](#)” on page 455 under the AFrame class.

## DeleteTextInsetContents

See [DeleteTextInsetContents](#) under the TiApiClient class.

## UpdateTextInset

See [UpdateTextInset](#) under the TiApiClient class.

## ConvertToText

### Description

This method is called upon a text inset (Examples: `TiApiClient`, `TiFlow`, `TiText`, `TiTextTable`) object and converts it to a text. This method deletes the text inset and removes the locked text range around it without modifying the content.

### Returns

Void

### Syntax

```
ConvertToText ()
```

## Delete

See “[Delete](#)” on page 453 under the AFrame class.

## DeletePropByName

See “[DeletePropByName](#)” on page 699 under the TiApiClient.

## TiText

An `FO_TiText` object represents text imported from a text file.

## GetText

See “[GetText](#)” on page 493 under the Cell class.

## GetProps

See “[GetProps](#)” on page 453 under the AFrame class.

## SetProps

See “[SetProps](#)” on page 454 under the AFrame class.

## ObjectValid

See “[ObjectValid](#)” on page 455 under the AFrame class.

## DeleteTextInsetContents

See [DeleteTextInsetContents](#) under the TiApiClient class.

## UpdateTextInset

See [UpdateTextInset](#) under the TiApiClient class.

## ConvertToText

### Description

This method is called upon a text inset (Examples: `TiApiClient`, `TiFlow`, `TiText`, `TiTextTable`) object and converts it to a text. This method deletes the text inset and removes the locked text range around it without modifying the content.

### Returns

Void

### Syntax

```
ConvertToText ()
```

## Delete

See “[Delete](#)” on page 453 under the AFrame class.

## DeletePropByName

See “[DeletePropByName](#)” on page 699 under the TiApiClient class.

## TiTextTable

An `FO_TiTextTable` object represents text imported from a text file into a table.

## GetText

See “[GetText](#)” on page 493 under the Cell class.

## GetProps

See “[GetProps](#)” on page 453 under the AFrame class.

## SetProps

See “[SetProps](#)” on page 454 under the AFrame class.

## ObjectValid

See “[ObjectValid](#)” on page 455 under the AFrame class.

## DeleteTextInsetContents

See [DeleteTextInsetContents](#) under the TiApiClient class.

## UpdateTextInset

See [UpdateTextInset](#) under the TiApiClient class.

## ConvertToText

### Description

This method is called upon a text inset (Examples: [TiApiClient](#), [TiFlow](#), [TiText](#), [TiTextTable](#)) object and converts it to a text. This method deletes the text inset and removes the locked text range around it without modifying the content.

### Returns

Void

### Syntax

```
ConvertToText ()
```

## Delete

See “[Delete](#)” on page 453 under the AFrame class.

## DeletePropByName

See “[DeletePropByName](#)” on page 699 under the TiApiClient.

## TypedVals

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

## concat

### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

### Returns

[TypedVal](#)

### Syntax

`concat (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<code>TypedVal</code>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

`TypedVal`

### Syntax

`pop ()`

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

`push (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<code>TypedVal</code>	No	The value to push into the array.

## UBytes

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



## concat

### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

### Returns

`UBytes`

### Syntax

`concat (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<code>UBytes</code>	No	Can also be arrays.

## pop

### Description

Removes the last element from the array.

### Returns

`uint`

### Syntax

`pop ()`

## push

### Description

Returns the new length of the array.

### Returns

number

### Syntax

`push (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	<code>uint</code>	No	The value to push into the array.

## Uints

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

### concat

#### Description

The original array is unchanged. If an array is provided as a parameter to `concat()`, each of its elements are appended as separate array elements at the end of the new array. Returns a new array, the result of concatenation the given values to the end of the original array.

#### Returns

`Uints`

#### Syntax

`concat (value)`

#### Parameters

Parameter name	Data Type	Optional	Description
value	<code>Uints</code>	No	Can also be arrays.

## pop

#### Description

Removes the last element from the array.

#### Returns

`uint`

#### Syntax

`pop ()`

## push

#### Description

Returns the new length of the array.

#### Returns

`number`

#### Syntax

`push (value)`

### Parameters

Parameter name	Data Type	Optional	Description
value	uint	No	The value to push into the array.

## UnanchoredFrame

### Delete

#### Description

Deletes the specified unanchored frame.

See [Delete](#) under the AFrame class for more information.

#### Returns

int

#### Syntax

```
Delete()
```

### GetProps

#### Description

The `GetProps()` method retrieves the properties of the specified unanchored frame.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

#### Syntax

```
GetProps()
```

### SetProps

#### Description

The `SetProps()` method sets the properties of the specified unanchored frame.

See [SetProps](#) under the AFrame class for more information.

#### Returns

Void

#### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the unanchored frame is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

`ObjectValid()`

## Var

## Delete

### Description

Deletes the specified var object.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

`Delete()`

## GetText

### Description

Gets the text from the var object.

See [GetText](#) under the Cell class for more information.

### Returns

[TextItems](#)

### Syntax

`GetText (flags)`

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified var object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the specified var object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the var object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

`ObjectValid()`

## VarFmt

### Delete

#### Description

Deletes the specified varfmt object.

See [Delete](#) under the AFrame class for more information.

#### Returns

`int`

### Syntax

`Delete()`

### GetProps

#### Description

The `GetProps()` method retrieves the properties of the specified varfmt object.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

### Syntax

`GetProps()`

### SetProps

#### Description

The `SetProps()` method sets the properties of the specified var object.

See [SetProps](#) under the AFrame class for more information.

#### Returns

`Void`

### Syntax

`SetProps(setVal)`

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the varfmt object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

`ObjectValid()`

## XRef

### Delete

### Description

Deletes the specified XRef object.

See [Delete](#) under the AFrame class for more information.

### Returns

int

### Syntax

`Delete()`

### GetText

### Description

Gets the text from the XRef object.

See [GetText](#) under the Cell class for more information.

### Returns

[TextItems](#)

### Syntax

`GetText (flags)`

### Parameters

Parameter name	Data Type	Optional	Description
flags	int	No	Denotes the various bit flags that specify the items to be retrieved.

## GetProps

### Description

The `GetProps()` method retrieves the properties of the specified XRef object.

See [GetProps](#) under the AFrame class for more information.

### Returns

[PropVals](#)

### Syntax

```
GetProps()
```

## SetProps

### Description

The `SetProps()` method sets the properties of the specified XRef object.

See [SetProps](#) under the AFrame class for more information.

### Returns

Void

### Syntax

```
SetProps(setVal)
```

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the xref object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int



### Syntax

`ObjectValid()`

## XRefFmt

### Delete

#### Description

Deletes the specified XRefFmt object.

See [Delete](#) under the AFrame class for more information.

#### Returns

`int`

### Syntax

`Delete()`

### GetProps

#### Description

The `GetProps()` method retrieves the properties of the specified XRefFmt object.

See [GetProps](#) under the AFrame class for more information.

#### Returns

[PropVals](#)

### Syntax

`GetProps()`

### SetProps

#### Description

The `SetProps()` method sets the properties of the specified XRefFmt object.

See [SetProps](#) under the AFrame class for more information.

#### Returns

`Void`

### Syntax

`SetProps(setVal)`

### Parameters

Parameter name	Data Type	Optional	Description
setVal	<a href="#">PropVals</a>	No	The property list.

## ObjectValid

### Description

Returns 1 if the XRefFmt object is valid, else returns 0.

See [ObjectValid](#) under the AFrame class for more information.

### Returns

int

### Syntax

`ObjectValid()`

## app

### Close

See “[Close](#)” on page 471 under the Book class.

## GetNamedObject

### Description

Gets the object with a specified name (`Name` property) and object type.

If an error occurs, this method returns the the object, or 0. If this method fails, the method assigns one of the following values to `FA_errno`:

Error	Meaning
<code>Constants.FE_BadDocId (-2)</code>	Invalid document.
<code>Constants.FE_TypeUnNamed (-19)</code>	Objects of the specified type do not have names.
<code>Constants.FE_NameNotFound (-20)</code>	Object with the specified name and type does not exist in the specified document.

### Returns

Object

### Syntax

`GetNamedObject(objType, name)`

### Parameters

Parameter name	Data Type	Optional	Description
objType	int	No	Type of the object to get.
name	string	No	Name of the object to get.

## GetNamedBook

### Description

Gets the book object with a specified name.

If this method fails, the method assigns the following value to `FA_errno`.

Error	Meaning
Constants.FE_NameNotFound (-20)	The book with the specified name does not exist.

### Returns

[Book](#)

### Syntax

`GetNamedBook (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the book object you want to get.

## GetNamedCommand

### Description

Gets a `FO_command` object with the specified command string.

If this method fails, the method assigns the following value to `FA_errno`.

Error	Meaning
Constants.FE_NameNotFound (-20)	The object with the specified name does not exist.

### Returns

[Command](#)

### Syntax

`GetNamedCommand (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the command object you want to get.

## GetNamedMenu

### Description

Gets a menu object with the specified menu string.

If this method fails, the method assigns the following value to `FA_erro`.

Error	Meaning
Constants.FE_NameNotFound (-20)	A menu with the specified name does not exist.

### Returns

[Menu](#)

### Syntax

`GetNamedMenu (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the menu object you want to get.

## GetNamedMenuItemSeparator

### Description

Gets a menu item separator object with the specified menu item separator string.

If this method fails, the method assigns the following value to `FA_erro`.

Error	Meaning
Constants.FE_NameNotFound (-20)	A menu item separator with the specified name does not exist.

### Returns

[MenuItemSeparator](#)

### Syntax

`GetNamedMenuItemSeparator (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the menu item separator you want to get.

## NewNamedObject

### Description

Creates a new objects and returns it.

If this method fails, the method assigns one of the following values to `FA_errno`.

Error	Meaning
<code>FE_BadName (-33)</code>	Specified name for the new object is invalid.
<code>FE_BadNew (-23)</code>	Object cannot be created.
<code>FE_DupName (-32)</code>	Specified name for the new object belongs to an existing object.

### Returns

Object

### Syntax

`NewNamedObject(objType, name)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>objType</code>	<code>int</code>	No	The type of the object you want to create.
<code>name</code>	<code>string</code>	No	The name of the object you want to create.

## NewNamedBook

### Description

Creates a new book objects and returns it.

If this method fails, the method assigns one of the following values to `FA_errno`.

Error	Meaning
<code>FE_BadName (-33)</code>	Specified name is invalid.
<code>FE_BadNew (-23)</code>	Object cannot be created.
<code>FE_DupName (-32)</code>	Specified name for the new object belongs to an existing object.

### Returns

[Book](#)

### Syntax

`NewNamedBook(name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	The name of the book object to create.

## NewNamedCommand

### Description

Creates a new command object and returns it

If this method fails, the method assigns one of the following values to `FA_erno`.

Error	Meaning
FE_BadName (-33)	Specified name is invalid.
FE_BadNew (-23)	Command cannot be created.
FE_DupName (-32)	Specified name for the new command belongs to an existing command.

### Returns

[Command](#)

### Syntax

`NewNamedCommand (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	Name of the new command object to create.

## NewNamedMenu

### Description

Creates a menu object with the specified name and returns it

If this method fails, the method assigns one of the following values to `FA_erno`.

Error	Meaning
FE_BadName (-33)	Specified name is invalid.
FE_BadNew (-23)	Menu cannot be created.
FE_DupName (-32)	Specified name for the new menu belongs to an existing menu.

### Returns

[Menu](#)

### Syntax

`NewNamedMenu (name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	Name of the menu object to create.

## NewNamedMenuItemSeparator

### Description

Creates a new menu item separator with the specified name and returns it

If this method fails, the method assigns one of the following values to `FA_erro`.

Error	Meaning
FE_BadName (-33)	Specified name is invalid.
FE_BadNew (-23)	Menu item separator cannot be created.
FE_DupName (-32)	Specified name for the new menu item separator belongs to an existing menu item separator.

### Returns

[MenuItemSeparator](#)

### Syntax

`NewNamedMenuItemSeparator(name)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	Name of the new menu item separator object.

## GetProps

See “[GetProps](#)” on page 453 under the `AFrame` class.

## SetProps

See “[SetProps](#)” on page 454 under the `AFrame` class.

## ObjectValid

See “[ObjectValid](#)” on page 455 under the `AFrame` class.

# global

## Alert

### Description

Displays an alert box with a message. Depending on the constant you specify for the `type` parameter, the alert displays either the OK and Cancel buttons, Yes and No buttons, or a Continue button.

The `type` parameter takes one of the following values:

Constant	Value
Constants.FF_ALERT_OK_DEFAULT (0)	Displays the OK and Cancel buttons; OK is the default.
Constants.FF_ALERT_CANCEL_DEFAULT (1)	Displays the OK and Cancel buttons; Cancel is the default.
Constants.FF_ALERT_CONTINUE_NOTE (2)	Displays the OK button.
Constants.FF_ALERT_CONTINUE_WARN (3)	Displays the OK button with a warning indicator.
Constants.FF_ALERT_YES_DEFAULT (4)	Displays the Yes and No buttons; Yes is the default.
Constants.FF_ALERT_NO_DEFAULT (5)	Displays the Yes and No buttons; No is the default.

The method returns 0 if the user clicked OK, Continue, or Yes.

The method returns -1 if the user clicked Cancel or No.

### Returns

int

### Syntax

Alert(message, type)

### Parameters

Parameter name	Data Type	Optional	Description
message	string	No	The message that appears in the dialog box. Messages longer than 255 characters, are truncated.
type	int	No	The dialog box type.

## Alive

### Description

Checks whether the current asynchronous client has a connection with a FrameMaker process. Call this method after registering the asynchronous client using `winConnectSession()`.

The method returns a positive integer if there is a current connection to a FrameMaker process. Otherwise it returns 0.

### Returns

int



### Syntax

Alive()

## CallClient

### Description

Allows a client to call another client. The method is useful for calling FrameMaker clients, such as the structure generator and the element catalog manager.

On success, the method returns `FE_Success` or the value specified by the target client's last call to `ReturnValue()`.

**Note:** Calls to the structure generator always return `FE_Success` no matter what string is passed to it as an argument.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_NameNotFound (-20)	There is no client with the specified name in the current FrameMaker productsession.
Constants.FE_BadParameter (-43)	For the TableSort client only: One of the arguments is invalid. For example, you gave a value for the sort key that is greater than the number of columns or rows in the current table selection, or you have no table cells selected.

### Returns

int

### Syntax

CallClient(cname, arg)

### Parameters

Parameter name	Data Type	Optional	Description
cname	string	No	The registered name of the target client.
arg	string	No	A string that is passed to the target client.

## CallClientEx

### Description

Allows a client to call another client. The method is useful for calling FrameMaker clients, such as the structure generator and the element catalog manager.

On success, the method returns `FE_Success` or the value specified by the target client's last call to `ReturnValue()`.

**Note:** Calls to the structure generator always return `FE_Success` no matter what string is passed to it as an argument.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_NameNotFound (-20)	There is no client with the specified name in the current FrameMaker productsession.

Error	Reason
Constants.FE_BadParameter (-43)	For the TableSort client only: One of the arguments is invalid. For example, you gave a value for the sort key that is greater than the number of columns or rows in the current table selection, or you have no table cells selected.

### Returns

TypedVal

### Syntax

CallClientEx(cname, arg)

### Parameters

Parameter name	Data Type	Optional	Description
cname	string	No	The registered name of the target client.
arg	string	No	A string that is passed to the target client.

## CheckStatus

### Description

Checks the scripts returned by `Open()`, `Import()`, `Save()`, and `UpdateBook()` to determine if a specified status bit is set. Returns 1 if the bit is set, else returns 0.

### Returns

int

### Syntax

CheckStatus(p, statusBit)

### Parameters

Parameter name	Data Type	Optional	Description
p	PropVals	No	The property list returned by <code>Open()</code> , <code>Save()</code> , <code>Import()</code> , or <code>UpdateBook()</code> .
statusBit	int	No	The status bit to test.

## ClientDir

### Description

Returns the name of the current ESTK client's directory.

**Note:** Use `Free()` to free the string returned by `ClientDir()` when you are done with it.

The method assigns the following value to `FA_errno`, on failure:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

### Returns

string

### Syntax

```
ClientDir()
```

## ClientName

### Description

Returns the registered name of the current client (the client that calls `ClientName()`).

*Note:* Use `Free()` to free the string returned by `ClientDir()` when you are done with it.

The method assigns the following value to `FA_errno`, on failure:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

### Returns

string

### Syntax

```
ClientName(_)
```

## CMSRegister

### Description

Registers a CMS client.

### Returns

CMS Object

### Syntax

```
CMSRegister (cmsName);
```

### Parameters

Parameter name	Data Type	Optional	Description
cmsName	int	No	Name of the CMS to register

If `CMSRegister()` fails, the method assigns following values to `FA_errno`:

**Function Summary**

Error	Reason
Constants.FE_CMSNameAlreadyRegistered	The method attempts to register a CMS that is already registered.
Constants.FE_BadParameter	The function call specified an invalid parameter.

**CMSGetCmsIdFromName****Description**

Gets the CMS registration id from CMS name.

**Returns**

CMS Object

**Syntax**

```
CMSGetCmsIdFromName (cmsName);
```

**Parameters**

Parameter name	Data Type	Optional	Description
cmsName	int	No	The Name of the CMS

If CMSGetCmsIdFromName() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadParameter	The function call specified an invalid parameter.

**CMSAddMenuEntry****Description**

Adds a custom menu entry in the context menu within the FrameMaker interface.

**Returns**

CMS Object

**Syntax**

```
CMSAddMenuEntry (menuId, menuEntry);
```

**Parameters**

Parameter name	Data Type	Optional	Description
menuId	int	No	The ID of the Parent menu
menuEntry	int	No	The F_CMSMenuItemT structure describes a custom menu definition

If CMSAddMenuEntry() fails, the method assigns following values to FA\_errno:

**Function Summary**

Error	Reason
Constants.FE_CMSBadObjectId	The client specified an invalid menu ID.
Constants.FE_BadParameter	The function call specified an invalid parameter.

**CustomDoc****Description**

Creates a new custom document using the FrameMaker product's default new document template.

The sidedness parameter can have one of the following values:

Vaule	Page Characteristics
Constants.FF_Custom_SingleSided (0)	Single sided.
Constants.FF_Custom_FirstPageRight (1)	Double-sided, starting with a right page.
Constants.FF_Custom_FirstPageLeft (2)	Double-sided, starting with a left page.

The method returns the new document on success.

The method assigns the following value to `FA_errno`, on failure:

Error	Reason
Constants.FE_WrongProduct (-60)	The product version does not support this operation.
Constants.FE_BadParameter (-43)	Parameter has an invalid value.

**Returns**

[Doc](#)

**Syntax**

```
CustomDoc(width, height, numCols, columnGap, topMargin, botMargin, leftinsideMargin,
rightoutsideMargin, sidedness, makeVisible)
```

**Parameters**

Parameter name	Data Type	Optional	Description
width	int	No	The document page width.
height	int	No	The document page height.
numCols	int	No	The default number of columns.
columnGap	int	No	The default column spacing.
topMargin	int	No	The document page top margin.
botMargin	int	No	The document page bottom margin.
leftinsideMargin	int	No	The left margin for single-sided documents, or the inside margin for double-sided documents.

Parameter name	Data Type	Optional	Description
rightoutsideMargin	int	No	The right margin for single-sided documents, or the outside margin for double-sided documents.
sidedness	int	No	A constant that specifies whether the document is single-sided or doublesided and on which side the document starts.
makeVisible	bool	No	Specifies whether the document is visible after it is created. Set to <code>True</code> to make the document visible.

## DefineMenu

### Description

Defines a menu (FO\_Menu object). After you define a menu, you can add it to a menu or a menu bar with `AddMenuToMenu()`.

If you call `DefineMenu()` and specify the name of a menu that is already defined in the user's menu configuration files, the FrameMaker product gives precedence to the definition in the configuration files. If the configuration files assign a label to the menu, the FrameMaker product uses it instead of the one you specify. If the user has already defined a menu with the name specified by `name`, `DefineMenu()` ignores the `label` parameter and uses the label specified by the user.

The method returns the new menu on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_WrongProduct (-60)	Current FrameMaker product does not support this operation or <code>fmbatch</code> is running
Constants.FE_BadParameter (-43)	Parameter has an invalid value.
Constants.FE_BadOperation (-27)	Parameters specify an invalid operation.
Constants.FE_NotMenu (-72)	<code>name</code> specifies a command; cannot redefine a command as a menu.
Constants.FE_SystemError (-42)	System error.

### Returns

[Menu](#)

### Syntax

`DefineMenu(name, label)`

### Parameters

Parameter name	Data Type	Optional	Description
name	string	No	A unique name for the menu. If the user or an ESTK client has already defined a command or menu with this name, the new menu replaces it.
label	string	No	The title of the menu as it appears on the menu bar or menu.

## DisconnectFromSession

### Description

Ends communication with a FrameMaker product process.

The method returns `FE_Success` if it succeeds, or a system error code if an error occurs.

### Returns

`int`

### Syntax

```
DisconnectFromSession()
```

## Err

### Description

Prints the client's name and a message to the console.

### Returns

`Void`

### Syntax

```
Err (message)
```

### Parameters

Parameter name	Data Type	Optional	Description
message	string	No	The message to print.

## FamilyFonts

### Description

Returns a `Fonts` object with the permutations of angles, variations, and weights available for a specified font family.

### Returns

`Fonts`

### Syntax

```
FamilyFonts (family)
```

### Parameters

Parameter name	Data Type	Optional	Description
family	int	No	The index of the font family (in the list of fonts in the session).

## GetKeyCatalog

### Description

Finds a key catalog with the specified 'tag'.

### Returns

KeyCatalog

### Syntax

```
GetKeyCatalog (tag) ;
```

### Parameters

Parameter name	Data Type	Optional	Description
tag	string	No	The tag of the new Key Catalog being created.

If GetKeyCatalog() fails, the method assigns following values to FA\_errno:

Error	Reason
Constants.FE_BadName	The tag provided is not valid or the key catalog with this tag does not exist.

## GetEncodingForFamily

### Description

Returns the encoding that FrameMaker uses for the font family.

Returns one of the following strings indicating the encoding for the font:

Value	Meaning
FrameRoman	Roman Text.
JISX0208.ShiftJIS	Japanese Text.
BIG5	Traditional Chinese Text.
GB2312-80.EUC	Simplified Chinese Text.
KSC5601-1992	Korean Text.
Multiple	More than one encoding for the font family.

If the returned string is Multiple, the font family includes variations that are represented by different encodings. You should use `ApiFamilyFonts()` to get a list of the variations for the family. Then you can use `GetEncodingForFont()` to get the encoding for a specific variation.

### Returns

string

### Syntax

```
GetEncodingForFamily (family)
```



### Parameters

Parameter name	Data Type	Optional	Description
family	int	No	The font family for which the encoding is to be retrieved.

## GetEncodingForFont

### Description

Returns the encoding that FrameMaker uses for a specific font with a specific combination of weight, angle, and variation.

Returns one of the following strings indicating the encoding for the font:

Value	Meaning
FrameRoman	Roman Text.
JISX0208.ShiftJIS	Japanese Text.
BIG5	Traditional Chinese Text.
GB2312-80.EUC	Simplified Chinese Text.
KSC5601-1992	Korean Text.

### Returns

string

### Syntax

GetEncodingForFont (font)

### Parameters

Parameter name	Data Type	Optional	Description
font	Font	No	

## GetImportDefaultParams

### Description

Gets a default property list that you can use to call `Import()`.

The method returns a `PropVals` structure containing the properties retrieved.

For more information see “[Import](#)” on page 479.

### Returns

`PropVals`

### Syntax

GetImportDefaultParams()

## GetOpenDefaultParams

### Description

Gets a default property list that you can use to call `Open()`.

### Returns

`PropVals`

### Syntax

```
GetOpenDefaultParams()
```

## GetPropIndex

### Description

Gets the index of a property-value pair (`PropVal` structure) within a property list. `GetPropIndex()` is a convenience routine that makes it easier to manipulate the properties in a property list.

The method returns the index (in the property list) of the property's `PropVal` structure, or `Constants.FE_BadPropNum (-4)` if an error occurs.

### Returns

`int`

### Syntax

```
GetPropIndex(propVals, propNum)
```

### Parameters

Parameter name	Data Type	Optional	Description
<code>propVals</code>	<code>PropVals</code>	No	The property list.
<code>propNum</code>	<code>int</code>	No	The property for which the index is to be retrieved.

## GetSaveDefaultParams

### Description

Gets a default property list that you can use to call `Save()`.

On success, the method returns a `PropVals` list as shown in the following table.

The first value listed next to each property is the value that `GetSaveDefaultParams()` assigns to the property. The other values are values that you can assign to the property as desired.

Property	Values
AlertUserAboutFailure	<p>Specifies whether to notify user if something unusual happens during the Save operation. The value is one of:</p> <ul style="list-style-type: none"> <li>• False: Do not notify user.</li> <li>• True: Notify user.</li> </ul>
AutoBackupOnSave	<p>Specifies whether to create a backup file. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_SaveUserPrefAutoBackup(2) - Follow preference specified by the session's AutoBackup property.</li> <li>• Constants.FV_SaveYesAutoBackup(0) - Make a backup.</li> <li>• Constants.FV_SaveNoAutoBackup(1) - Do not make a backup.</li> </ul>
DontNotifyAPIClients	<p>Specifies whether to save the file without notifying other clients. The value is one of:</p> <ul style="list-style-type: none"> <li>• False - Notify other clients.</li> <li>• True - Do not notify other clients.</li> </ul>
FileType	<p>Specifies the type of file to save to. This file type must be one that FrameMaker saves natively. Note that HTML and XML are saved via filters, and so you must specify a filter hint string via Constants.FS_SaveFileTypeHint(16).</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_SaveFmtBinary(0) - Save in Frame binary format for this version of FrameMaker.</li> <li>• Constants.FV_SaveFmtBinary60(11) - Save in binary format for FrameMaker 6.0.</li> <li>• Constants.FV_SaveFmtInterchange(1) - Save as MIF.</li> <li>• Constants.FV_SaveFmtSgml(7) - Save in SGML format.</li> <li>• Constants.FV_SaveFmtText(6) - Save in Text Only format.</li> <li>• Constants.FV_SaveFmtXml(10) - Save in XML format.</li> <li>• Constants.FV_SaveFmtPdf(9) - Save as PostScript, and then invoke Acrobat Distiller to create a PDF version of the document. This is the same as choosing PDF from the Format popup menu in the Save As dialog box.</li> <li>• Constants.FV_SaveFmtFilter(8) - Filter on save, using Constants.FS_SaveFileTypeHint(16) to determine the filter.</li> </ul>
FilesInUse	<p>Another user or session is recorded in the file's lock file. The value is one of:</p> <ul style="list-style-type: none"> <li>• Constants.FV_DoCancel(0) - Cancel the Save operation.</li> <li>• Constants.FV_DoShowDialog(4) - Display the File In Use dialog box.</li> <li>• Constants.FV_ResetLockAndContinue(7) - Attempt to reset the file lock and save the document.</li> </ul>

Property	Values
LockCantBeReset	<p>The user clicked Save Anyway in the File In Use dialog box, or the value of <code>Constants.FS_FileInUse (TBD)</code> is set to <code>Constants.FV_ResetLockAndContinue (7)</code>, but the lock file cannot be reset. This is usually due to permissions in the lock file.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_DoCancel(0)</code> - Cancel the Save operation.</li> <li>• <code>Constants.FV_DoShowDialog(4)</code> - Display the Cannot Lock File dialog box.</li> <li>• <code>Constants.FV_DoOK(1)</code> - Save the document anyway.</li> </ul>
ModDateChanged	<p>The file has changed since the last time it was opened or saved in the current session. Somebody else has probably modified the file.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_DoCancel(0)</code> - Cancel the Save operation.</li> <li>• <code>Constants.FV_DoShowDialog(4)</code> - Display the File Has Changed alert box.</li> <li>• <code>Constants.FV_DoOK(1)</code> - Save the document anyway.</li> </ul>
SaveFileNotWritable	<p>The file permissions will not allow the file to be saved.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_DoCancel(0)</code> - Cancel the Save operation.</li> <li>• <code>Constants.FV_DoShowDialog(4)</code> - Display the Cannot Lock File alert box.</li> </ul>
SaveFileTypeHint	<p>If <code>Constants.FS_FileType (TBD)</code> is <code>Constants.FV_SaveFmtFilter(8)</code>, this string enables the FrameMaker product to call the correct filter. For example, use <code>0001ADBEHTML</code> to save as HTML or <code>0001ADBEXML</code> to save as XML.</p>
MakePageCount	<p>Specifies how to round the page count.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_UseCurrentSetting(0)</code> - Use default specified by the document property, <code>PageRounding</code>.</li> <li>• <code>Constants.FV_DontChangePageCount(1)</code> - Leave pages as is.</li> <li>• <code>Constants.FV_MakePageCountEven(2)</code> - With odd number of pages, add a page to end of document.</li> <li>• <code>Constants.FV_MakePageCountOdd(3)</code> - With even number of pages, add a page to end of document.</li> <li>• <code>Constants.FV_DeleteEmptyPages(4)</code> - Remove extra pages at end of document.</li> </ul>
RetainNameStripe	<p>Specifies whether to change the name in document title bar to the name to which the file is saved.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>False</code> - Change the name in the title bar to the name in which the file is saved.</li> <li>• <code>True</code> - Do not change the name in the title bar.</li> </ul>

Property	Values
SaveAsModeName	<p>Specifies where to get filename if <code>Constants.FS_SaveMode(4)</code> is set to <code>Constants.FV_ModeSaveAs(1)</code>.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_SaveAsNameProvided(2)</code> - Save under the filename specified in the <code>saveAsName</code> parameter of <code>Save()</code>.</li> <li>• <code>Constants.FV_SaveAsUseFileName(1)</code> - Save as name shown on the document title bar.</li> <li>• <code>Constants.FV_SaveAsNameAskUser(0)</code> - Prompt user for name.</li> </ul>
SaveMode	<p>Specifies whether to use Save or Save As mode.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_ModeSaveAs(1)</code> - Use Save As mode.</li> <li>• <code>Constants.FV_ModeSave(0)</code> - Use Save mode.</li> </ul>
SaveTextExtraBlankLineAtEOP	<p>Specifies whether to add an extra line at the end of each paragraph if the file is being saved as Text Only.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>False</code> - Do not add an extra line.</li> <li>• <code>True</code> - Add an extra line.</li> </ul>
SaveTextTblSetting	<p>Specifies how to deal with tables if the file is being saved as Text Only.</p> <p>The value is one of:</p> <ul style="list-style-type: none"> <li>• <code>Constants.FV_SaveTblUserPref(0)</code> - Use setting last specified in Save as Text dialog box.</li> <li>• <code>Constants.FV_SaveTblRowsAsPgfs(1)</code> - Save each table cell as a paragraph row-by-row.</li> <li>• <code>Constants.FV_SaveTblColsAsPgfs(2)</code> - Save each table cell as a paragraph column-by-column.</li> <li>• <code>Constants.FV_SaveSkipTbIs(3)</code> - Omit tables from a Text Only file.</li> <li>• <code>Constants.FV_SaveTextTblCellSeparator(TBD)</code> - The character to write as a cell separator in the text file.</li> <li>• <code>Constants.FV_SaveTextTblRowColumnSeparator(TBD)</code> - The character to write as a row or column separator in the text file.</li> </ul>
SgmlSaveApplication	<p>Retained for compatibility. Use the <code>StructuredSaveApplication</code> property instead.</p>
StructuredSaveApplication	<p>The value is one of:</p> <ul style="list-style-type: none"> <li>• String specifying the name of the structure application to use when saving a structured document as XML or SGML. This parameter takes precedence over any other structure application specification. If the specified application does not exist, the calling function will fail.</li> <li>• <code>Null</code> - No application used.</li> </ul>

Property	Values
ShowSaveTextDialog	Specifies whether to display dialog box if the file is being saved in Text Only format.  The value is one of: <ul style="list-style-type: none"><li>• False - Do not display dialog box.</li><li>• True - Display dialog box asking user whether to put paragraph returns at the end of each line.</li></ul>
UpdateFRVList	Specifies whether the file will be added to the list of files recently visited that appears in the File menu. This is set to False by default.  The value is one of: <ul style="list-style-type: none"><li>• False - Do not add the file to the list.</li><li>• True - Add the file to the list.</li></ul>

On failure, the method sets the `len` field of the returned structure to 0.

### Returns

[PropVals](#)

### Syntax

`GetSaveDefaultParams()`

## GetSupportedEncodings

### Description

Returns the font encodings supported for the current session.

The following strings indicate the encoding for fonts:

Value	Meaning
FrameRoman	Roman Text.
JISX0208.ShiftJIS	Japanese Text.
BIG5	Traditional Chinese Text.
GB2312-80.EUC	Simplified Chinese Text.
KSC5601-1992	Korean Text.

### Returns

[Strings](#)

### Syntax

`GetSupportedEncodings()`

## GetUpdateBookDefaultParams

### Description

TBD

### Returns

[PropVals](#)

### Syntax

```
GetUpdateBookDefaultParams ()
```

## GetNewXMLDefaultParams

### Description

Generates default open-parameters for NewXML().

### Returns

[PropVals](#)

### Syntax

```
GetNewXMLDefaultParams ();
```

A property list (PropVals) with the properties shown in the following table.

Property	Instruction or situation and possible values
Constants.FS_Doctype	Specifies a doctype to be used for creating a new XML document
Constants.FS_PublicId	Specifies a public id to be used for creating a new XML document
Constants.FS_SystemId	Specifies a DTD-system id to be used for creating a new xml document
Constants.FS_Extension	Used to provide a custom extension for the new XML document (like Untitled1.dita). Without customization, FrameMaker determines the extension by itself based on file type.
Constants.FS_Visible	A boolean property that indicates if the new XML document shall be visible or hidden.

## IsEncodingSupported

### Description

Checks whether the specified encoding is supported for the current session. For example, unless FrameMaker is running on a system that supports Japanese text, Japanese encoding is not supported.

The `encodingName` parameter can be set to one of the following values:

Value	Meaning
FrameRoman	Roman Text.
JISX0208.ShiftJIS	Japanese Text.
BIG5	Traditional Chinese Text.

Value	Meaning
GB2312-80.EUC	Simplified Chinese Text.
KSC5601-1992	Korean Text.

Returns `True` if the specified encoding is supported for the current session, else returns `False`.

### Returns

`bool`

### Syntax

`IsEncodingSupported(encodingName)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>encodingName</code>	<code>string</code>	No	The encoding that you want to test.

## LoadMenuCustomizationFile

### Description

Loads a menu customization file.

A menu customization file is a text file containing statements that change the menus and commands the user sees in FrameMaker. For example, a menu customization file can change the name of a command or move a command from one menu to another.

The method returns `FE_Success` on success.

On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_WrongProduct (-60)</code>	The product version does not support this operation.
<code>Constants.FE_BadParameter (-43)</code>	Parameter has an invalid value.
<code>Constants.FE_BadOperation (-27)</code>	Parameters specify an invalid operation.
<code>Constants.FE_SystemError (-42)</code>	System error.

### Returns

`Void`

### Syntax

`LoadMenuCustomizationFile(pathname, silent)`



**Parameters**

Parameter name	Data Type	Optional	Description
pathname	string	No	The pathname of the menu customization file to load. If you specify only a filename, the function looks in the client directory. If <code>silent</code> is set to <code>False</code> , the pathname specified by <code>pathname</code> is used as the default in the Menu Customization File dialog box.
silent	bool	No	Specifies whether to display the Menu Customization File dialog box and allow the user to choose the file.  To display the dialog box and allow the user to choose the file, specify <code>False</code> .  To use the file specified by <code>pathname</code> without asking the user, specify <code>True</code> .

**NewKeyCatalog**

**Description**

Creates a new key catalog with the specified 'tag'.

**Returns**

KeyCatalog

**Syntax**

`NewKeyCatalog (tag) ;`

**Parameters**

Parameter name	Data Type	Optional	Description
tag	string	No	The tag of the new Key Catalog being created.

If `NewKeyCatalog()` fails, the method assigns following values to `FA_erro`:

Error	Reason
<code>Constants.FE_BadName</code>	The tag provided is not valid.
<code>Constants.FE_DupName</code>	A key catalog for the tag provided already exists.

**NewXML**

**Description**

Creates a new, untitled XML.

**Returns**

Doc

**Syntax**

`ApiNewXML (opennewXMLParams, newXMLReturnParams) ;`

**Parameters**

Parameter name	Data Type	Optional	Description
opennewXMLParamsp	PropVals	No	A property list telling the FrameMaker product how to open the file and how to respond to errors and other conditions. To use the default list, specify NULL.
newXMLOpenReturnParams	PropVals	No	A property list that returns the filename and provides information about how the FrameMaker product opened the file. It must be initialized before you call F_ApiNewXMLOpen().

To get a property list to specify for the newXMLParams parameter, use GetNewXMLDefaultParams.

**Notification**

**Description**

Requests that the FrameMaker product notify your client whenever a specified event, or stage of an event, occurs.

*Note: If FrameMaker encounters an internal error and exits, it does not send any notification to your client about operations performed after the error occurred. For example, after an error, FrameMaker allows the user to save changes in open documents, but it does not notify any clients of the save operations.*

Many events have several notification points or stages for which you can request notification. The following table lists the notification points and the constants that specify them:

Event	Notification Point	Notification Constant
Frame binary document opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li>• After checking the type of file to be opened</li> <li>• Before opening the file</li> <li>• After opening the file</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li>• Constants.FA_Note_PostFileType (10)</li> <li>• Constants.FA_Note_PreOpenDoc (1)</li> <li>• Constants.FA_Note_PostOpenDoc (2)</li> </ul>
MIF document opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li>• After checking the type of file to be opened</li> <li>• Before opening the file</li> <li>• After opening the file</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li>• Constants.FA_Note_PostFileType (10)</li> <li>• Constants.FA_Note_PreOpenMIF (3)</li> <li>• Constants.FA_Note_PostOpenMIF (4)</li> </ul>
SGML document opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li>• After checking the type of file to be opened</li> <li>• Before opening the file</li> <li>• After opening the file</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li>• Constants.FA_Note_PostFileType (10)</li> <li>• Constants.FA_Note_PreOpenSGML (82)</li> <li>• Constants.FA_Note_PostOpenSGML (83)</li> </ul>

<b>Event</b>	<b>Notification Point</b>	<b>Notification Constant</b>
XML document opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li>• After checking the type of file to be opened</li> <li>• Before opening the file</li> <li>• After opening the file</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li>• Constants.FA_Note_PostFileType (10)</li> <li>• Constants.FA_Note_PreOpenXML (95)</li> <li>• Constants.FA_Note_PostOpenXML (96)</li> </ul>
Filterable document opened	Before checking the type of the file to be opened	Constants.FA_Note_FilterIn (14)
Frame binary book opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li>• After checking the type of file to be opened</li> <li>• Before opening the file</li> <li>• After opening the file</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li>• Constants.FA_Note_PostFileType (10)</li> <li>• Constants.FA_Note_PreOpenBook (16)</li> <li>• Constants.FA_Note_PostOpenBook (17)</li> </ul>
MIF book opened	<ul style="list-style-type: none"> <li>• Before checking the type of file to be opened</li> <li>• After checking the type of file to be opened</li> <li>• Before opening the file</li> <li>• After opening the file</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFileType (9)</li> <li>• Constants.FA_Note_PostFileType (10)</li> <li>• Constants.FA_Note_PreOpenBookMIF (18)</li> <li>• Constants.FA_Note_PostOpenBookMIF (19)</li> </ul>
User double-clicked to open a document in a book window	<ul style="list-style-type: none"> <li>• Before opening the file</li> <li>• After opening the file</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreBookComponentOpen (76)</li> <li>• Constants.FA_Note_PostBookComponentOpen (77)</li> </ul>
Generating a list or TOC for a document or a book	<ul style="list-style-type: none"> <li>• Before generating the file</li> <li>• After generating the file</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreGenerate (78)</li> <li>• Constants.FA_Note_PostGenerate (79)</li> </ul>
Document saved in Frame binary format	<ul style="list-style-type: none"> <li>• Before saving the document</li> <li>• After saving the document</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreSaveDoc (5)</li> <li>• Constants.FA_Note_PostSaveDoc (6)</li> </ul>
Document saved as MIF	<ul style="list-style-type: none"> <li>• Before saving the file as MIF</li> <li>• After saving the file as MIF</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreSaveMIF (7)</li> <li>• Constants.FA_Note_PostSaveMIF (8)</li> </ul>
Document saved as SGML	<ul style="list-style-type: none"> <li>• Before saving the file as SGML</li> <li>• After saving the file as SGML</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreSaveSGML (99)</li> <li>• Constants.FA_Note_PostSaveSGML (100)</li> </ul>
Document saved as XML	<ul style="list-style-type: none"> <li>• Before saving the file as XML</li> <li>• After saving the file as XML</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreSaveXML (97)</li> <li>• Constants.FA_Note_PostSaveXML (98)</li> </ul>

**Function Summary**

<b>Event</b>	<b>Notification Point</b>	<b>Notification Constant</b>
Document saved as PDF	<ul style="list-style-type: none"> <li>• Before specifying Acrobat settings and generating PostScript</li> <li>• After specifying Acrobat settings and generating PostScript</li> <li>• Before distilling the postscript</li> <li>• After distilling the postscript</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreSaveAsPDFDialog (71)</li> <li>• Constants.FA_Note_PostSaveAsPDFDialog (72)</li> <li>• Constants.FA_Note_PreDistill (73)</li> <li>• Constants.FA_Note_PostDistill (74)</li> </ul>
Document saved as filterable type	Before the document is saved	Constants.FA_Note_FilterOut (15)
Document exited	<ul style="list-style-type: none"> <li>• Before exiting the document</li> <li>• After exiting the document</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreQuitDoc (11)</li> <li>• Constants.FA_Note_PostQuitDoc (40)</li> </ul>
Book exited	<ul style="list-style-type: none"> <li>• Before exiting the book</li> <li>• After exiting the book</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreQuitBook (24)</li> <li>• Constants.FA_Note_PostQuitBook (41)</li> </ul>
First change made to a document since it was opened or saved	After the document is changed	Constants.FA_Note_DirtyDoc (12)
First change made to a book since it was opened or saved	After the book is changed	Constants.FA_Note_DirtyBook (25)
Book saved in Frame binary format	<ul style="list-style-type: none"> <li>• Before saving the book</li> <li>• After saving the book</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreSaveBook (20)</li> <li>• Constants.FA_Note_PostSaveBook (21)</li> </ul>
Book saved in MIF format	<ul style="list-style-type: none"> <li>• Before saving the MIF file</li> <li>• After saving the MIF file</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreSaveBookMIF (22)</li> <li>• Constants.FA_Note_PostSaveBookMIF (23)</li> </ul>
Document saved with Autosave	<ul style="list-style-type: none"> <li>• Before saving the document</li> <li>• After saving the document</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreAutoSaveDoc (32)</li> <li>• Constants.FA_Note_PostAutoSaveDoc (33)</li> </ul>
Document reverted	<ul style="list-style-type: none"> <li>• Before reverting the document</li> <li>• After reverting the document</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreRevertDoc (28)</li> <li>• Constants.FA_Note_PostRevertDoc (29)</li> </ul>
Book reverted	<ul style="list-style-type: none"> <li>• Before reverting the book</li> <li>• After reverting the book</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreRevertBook (30)</li> <li>• Constants.FA_Note_PostRevertBook (31)</li> </ul>
FrameMaker product exited	<ul style="list-style-type: none"> <li>• Before the OK to Exit dialog box appears</li> <li>• Immediately before exiting the session</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreQuitSession (26)</li> <li>• Constants.FA_Note_PostQuitSession (27)</li> </ul>
Another client calls <code>CallClient()</code> with <code>cname</code> set to the current client's name	After the call has been made to <code>CallClient()</code>	Constants.FA_Note_ClientCall (13)
Any user action, such as a command choice or text entry	After the FrameMaker product finishes processing the user action	Constants.FA_Note_BackToUser (34)
User clicks on Text inset owned by the current client	After the user clicked the inset	Constants.FA_Note_DisplayClientTIDialog (35)

**Function Summary**

<b>Event</b>	<b>Notification Point</b>	<b>Notification Constant</b>
FrameMaker product updates all text insets	When the client needs to update insets that belong to it	Constants.FA_Note_UpdateAllClientTi (36)
FrameMaker product updates a specific text inset	When the client needs to update a specified inset	Constants.FA_Note_UpdateClientTi (37)
Text or graphic imported	<ul style="list-style-type: none"> <li>• Before importing the text or graphic</li> <li>• After importing the text or graphic</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreImport (38)</li> <li>• Constants.FA_Note_PostImport (39)</li> </ul>
FrameMaker product command invoked or text entered in a document r	<ul style="list-style-type: none"> <li>• Before the FrameMaker product executes a command or adds text to the document</li> <li>• After the FrameMaker product executes a command or adds text to the document</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreFunction (42)</li> <li>• Constants.FA_Note_PostFunction (43)</li> </ul>
Mouse button clicked	<ul style="list-style-type: none"> <li>• Before the FrameMaker product responds to the mouse click</li> <li>• After the FrameMaker product responds to the mouse click</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreMouseCommand (44)</li> <li>• Constants.FA_Note_PostMouseCommand (45)</li> </ul>
Hypertext command invoked	<ul style="list-style-type: none"> <li>• Before the FrameMaker product executes a hypertext command</li> <li>• After the FrameMaker product executes a hypertext command</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreHypertext (46)</li> <li>• Constants.FA_Note_PostHypertext (47)</li> </ul>
The user clicked Go To Source in the cross reference dialog box	<ul style="list-style-type: none"> <li>• Before the FrameMaker product goes to the cross reference source</li> <li>• After the FrameMaker product goes to the cross reference source</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreGoToXrefSrc (80)</li> <li>• Constants.FA_Note_PostGoToXrefSrc (81)</li> </ul>
Document or book printed	<ul style="list-style-type: none"> <li>• After the user clicks OK in the Print dialog box, but before the FrameMaker product prints the document or book</li> <li>• After the FrameMaker product prints the document or book</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PrePrint (48)</li> <li>• Constants.FA_Note_PostPrint (49)</li> </ul>
Body page added to a document	After the FrameMaker product adds the body page	Constants.FA_Note_BodyPageAdded (50)
Body page deleted from a document	After the FrameMaker product deletes the body page	Constants.FA_Note_BodyPageDeleted (51)
Structural element inserted	<ul style="list-style-type: none"> <li>• Before the element is inserted</li> <li>• After the element is inserted</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreInsertElement (52)</li> <li>• Constants.FA_Note_PostInsertElement (53)</li> </ul>
Structural element copied	<ul style="list-style-type: none"> <li>• Before the element is copied</li> <li>• After the element is copied</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreCopyElement (60)</li> <li>• Constants.FA_Note_PostCopyElement (61)</li> </ul>
Structural element changed	<ul style="list-style-type: none"> <li>• Before the element is changed</li> <li>• After the element is changed</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreChangeElement (54)</li> <li>• Constants.FA_Note_PostChangeElement (55)</li> </ul>
Structural element wrapped	<ul style="list-style-type: none"> <li>• Before the element is wrapped</li> <li>• After the element is wrapped</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreWrapElement (56)</li> <li>• Constants.FA_Note_PostWrapElement (57)</li> </ul>

**Function Summary**

Event	Notification Point	Notification Constant
Structural element dragged	<ul style="list-style-type: none"> <li>• Before the element is dragged</li> <li>• After the element is dragged</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreDragElement (58)</li> <li>• Constants.FA_Note_PostDragElement (59)</li> </ul>
An attribute value is set	<ul style="list-style-type: none"> <li>• Before the attribute value is set</li> <li>• After the attribute value is set</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreSetAttrValue (62)</li> <li>• Constants.FA_Note_PreSetAttrValue (63)</li> </ul>
Element definitions are imported	<ul style="list-style-type: none"> <li>• Before the element definitions are imported</li> <li>• After the element definitions are imported</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreImportElemDefs (64)</li> <li>• Constants.FA_Note_PostImportElemDefs (65)</li> </ul>
Inline input of double-byte text	<ul style="list-style-type: none"> <li>• Before the text entry</li> <li>• After the text entry</li> </ul>	<ul style="list-style-type: none"> <li>• Constants.FA_Note_PreInlineTypeIn (69)</li> <li>• Constants.FA_Note_PostInlineTypeIn (70)</li> </ul>
Filter a file on import or export via a filter to file filter	A file to file filter has been invoked—this notification occurs before the file is imported	Constants.FA_Note_FilterFileToFile (75)

The notification constants are numbered sequentially, starting with 0. The method provides a constant, `Constants.FA_Note_Num (128)`, that specifies the total number of notifications. This makes it easy to request notification for all notification points.

The method returns `FE_Success` on success. On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
<code>Constants.FE_Transport (-1)</code>	A transport error occurred.
<code>Constants.FE_BadNotificationNum (-30)</code>	The specified notification number was invalid.

**Returns**

`int`

**Syntax**

```
Notification(notification, state)
```

**Parameters**

Parameter name	Data Type	Optional	Description
<code>notification</code>	<code>int</code>	No	Constant that specifies the notification point.
<code>state</code>	<code>int</code>	No	Specifies whether to turn notification on or off. 1 turns it on, and 0 turns it off

**PopClipboard****Description**

Pops the Clipboard stack, moving the entry on the top of the stack to the Clipboard.

The method returns `FE_Success` on success. On failure, the method sets `FA_errno` to one of the following values:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.
Constants.FE_BadOperation (-27)	Clipboard stack is empty.

**Returns**

int

**Syntax**

PopClipboard()

**Parameters**

Parameter name	Data Type	Optional	Description
_	Void	No	NULL

**PrintFAErrno**

**Description**

Prints the current error status, represented by the global variable, `FA_errno`. The method is useful for debugging clients.

When a method fails, it stores an error code in the global variable, `FA_errno`.

`FA_errno` retains the error code until another function fails and sets it or until your code explicitly sets it.

To determine whether a method call has failed, initialize `FA_errno` to `FE_Success` once before calling the method and check it once after the method call is completed. For example, if you call the `Notification()` method and specify an invalid notification constant, the method sets `FA_errno` to `Constants.FE_BadNotificationNum (-30)`. If you subsequently call `PrintFAErrno()`, it prints the string `FE_BadNotificationNum`.

On failure, `PrintFAErrno()` assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

**Returns**

Void

**Syntax**

PrintFAErrno(\_)

**Parameters**

Parameter name	Data Type	Optional	Description
_	Void	No	NULL

## PrintImportStatus

### Description

Prints status flags returned by `Import()`. It is useful for debugging your clients.

### Returns

Void

### Syntax

`PrintImportStatus(p)`

### Parameters

Parameter name	Data Type	Optional	Description
p	<a href="#">PropVals</a>	No	The property list that <code>Import()</code> returns in <code>importReturnParamspp</code> .

## PrintOpenStatus

### Description

Prints status flags returned by `Open()`. It is useful for debugging your clients.

### Returns

Void

### Syntax

`PrintOpenStatus(p)`

### Parameters

Parameter name	Data Type	Optional	Description
p	<a href="#">PropVals</a>	No	The property list that <code>Open()</code> returns in <code>openReturnParamspp</code> .

## PrintPropVal

### Description

Prints the value of a specified property. It is useful for debugging your clients.

### Returns

Void

### Syntax

`PrintPropVal(p)`



### Parameters

Parameter name	Data Type	Optional	Description
p	<a href="#">PropVal</a>	No	The property to print.

## PrintPropVals

### Description

Prints the values in a specified property list. It is useful for debugging your clients.

### Returns

Void

### Syntax

```
PrintPropVals(p)
```

### Parameters

Parameter name	Data Type	Optional	Description
p	<a href="#">PropVals</a>	No	The property to print.

## PrintSaveStatus

### Description

Prints errors returned by `save()`. It is useful for debugging your clients.

### Returns

Void

### Syntax

```
PrintSaveStatus(p)
```

### Parameters

Parameter name	Data Type	Optional	Description
p	<a href="#">PropVals</a>	No	The property list that <code>save()</code> returns in <code>saveReturnParamspp</code> .

## PrintTextItem

### Description

Prints the text in a specified text item. It is useful for debugging clients.

### Returns

Void

### Syntax

```
PrintTextItem(textItem)
```

### Parameters

Parameter name	Data Type	Optional	Description
textItem	<a href="#">TextItem</a>	No	The text item to print.

## PrintTextItems

### Description

Prints the text in a specified set of text items (`TextItems` structure). It is useful for debugging clients.

### Returns

Void

### Syntax

```
PrintTextItems(textItems)
```

### Parameters

Parameter name	Data Type	Optional	Description
textItems	<a href="#">TextItems</a>	No	The set of text items to print.

## PrintUpdateBookStatus

### Description

Prints errors returned by `UpdateBook()`. It is useful for debugging your clients.

### Returns

Void

### Syntax

```
PrintUpdateBookStatus(p)
```

### Parameters

Parameter name	Data Type	Optional	Description
p	<a href="#">PropVals</a>	No	The property list that <code>UpdateBook()</code> returns in <code>updateReturnParamspp</code> .

## PushClipboard

### Description

Pushes the current Clipboard contents onto the Clipboard stack. The method is useful if you want to use Clipboard functions, such as `Copy()` or `Cut()`, without losing the Clipboard's original contents.

On failure, the method assigns the following value to `FA_erno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

**Returns**

Void

**Syntax**

`PushClipboard()`

**Parameters**

Parameter name	Data Type	Optional	Description
	Void	No	NULL

**ReturnValue**

**Description**

Sets a return value for a client-defined callback. The method allows a client to provide status information to the FrameMaker product or client that called the callback. You can call this method in the following callbacks:

- `DialogEvent()`
- `Notify()`

`ReturnValue()` is useful for canceling FrameMaker product operations. When your client receives a `Constants.FA_PreNotificationPoint` notification for an operation, it can cancel the operation by calling `ReturnValue()` with `retvalue` set to `Constants.FR_CancelOperation (-10001)`.

For example, if your client's `Notify()` callback responds to all `Constants.FA_Note_PrePrint (48)` notifications by calling `ReturnValue()` with `retvalue` set to `Constants.FR_CancelOperation (-10001)`, FrameMaker cancels all print operations.

Your client can also call `ReturnValue()` in a `DialogEvent()` callback to prevent FrameMaker from closing a modal dialog box.

You can set `retvalue` to any integer. If you client sets `retval` in response to one of the following notifications, it can use the following listed constants:

Notification	Values that client can pass to <code>ReturnValue()</code>	Meaning
<code>Constants.FA_NOTE_PreNotificationPoint</code>	<code>Constants.FR_CancelOperation (-10001)</code>	Cancel the operation for which the notification was issued
<code>Constants.FA_Note_ClientCall (13)</code>	Any value recognized by the client that called <code>CallClient()</code> .	Client-defined.
<code>Constants.FA_Note_FilterIn (14)</code>	<ul style="list-style-type: none"> <li>• The document into which the file was filtered.</li> <li>• 0</li> </ul>	<ul style="list-style-type: none"> <li>• The document was filtered successfully.</li> <li>• The document was not filtered successfully.</li> </ul>

Notification	Values that client can pass to ReturnValue()	Meaning
Constants.FA_Note_DisplayClientTiDialog (35)	Constants.FR_DisplayedTiDialog (-10002)	The client has displayed its version of the Text Inset Properties dialog box.
Constants.FA_Note_PreSaveAsPDFDialog (71)	<ul style="list-style-type: none"> <li>• Constants.FR_CancelOperation (-10001)</li> <li>• Constants.FR_SkipStep (-10004)</li> </ul>	<ul style="list-style-type: none"> <li>• Cancel the Save As PDF operation.</li> <li>• Do not display the Acrobat Settings dialog box</li> </ul>
Constants.FA_Note_PostSaveAsPDFDialog (72)	Constants.FR_CancelOperation (-10001)	Cancel the Save as PDF operation.  <i>Note: This event occurs before the distilling operation begins. You can now cancel the peration after the user closes Save As PDF dialog box.</i>
Constants.FA_Note_PreDistill (73)	Constants.FR_CancelOperation (-10001)	Cancel the Save as PDF operation.

Calling `ReturnValue()` for notifications other than those listed above, has no effect.

A client can also call `ReturnValue()` in a `DialogEvent()` callback that responds to actions in a client-defined modal dialog box.

Normally, when the user clicks a button in a client-defined modal dialog box, FrameMaker calls the client's `DialogEvent()` callback and then closes the dialog box. However, if the client's `DialogEvent()` callback calls `ReturnValue()` with `retvalue` set to `Constants.FR_DialogStayUp (-10000)`, FrameMaker does not close the dialog box.

The following table lists the values that a client can pass to `ReturnValue()` in a `DialogEvent()` callback:

Value	Meaning
Constants.FR_DialogStayUp (-10000)	Do not close the modal dialog box in which the event occurred.
Any other value	Close the modal dialog box.

On success, `ReturnValue()` returns the value of the `retvalue` parameter that was set, the previous time `ReturnValue()` was called in the current callback function.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

**Returns**

int

**Syntax**

`ReturnValue (retvalue)`

**Parameters**

Parameter name	Data Type	Optional	Description
retvalue	int	No	The value to return.

## ShutDown

### Description

Closes an ESTK client's connection.

### Returns

Void

### Syntax

ShutDown(\_)

### Parameters

Parameter name	Data Type	Optional	Description
_	Void	No	NULL

## SimpleNewDoc

### Description

Creates a new document from a specified template.

*Note: If you call `SimpleNewDoc()` with `interactive` set to `True` and the user clicks either `Portrait`, `Custom`, or `Landscape` in the `New` dialog box, `SimpleNewDoc()` does not create a document. Instead, it returns 0 and sets `FA_errno` to either `Constants.FE_WantsPortrait` (-48), `Constants.FE_WantsCustom` (-46), or `Constants.FE_WantsLandscape` (-47), respectively. It is up to your client to create a portrait, custom, or landscape document.*

*Note: For more information on creating custom documents, see “[CustomDoc](#)” on page 724.*

If `SimpleNewDoc()` fails, the method assigns an error code to `FA_errno`. The error codes for `SimpleNewDoc()` are the same as those for `Open()`. For a list of these error codes, see “[Open](#)” on page 755.

### Returns

[Doc](#)

### Syntax

`SimpleNewDoc(templateName, interactive)`

### Parameters

Parameter name	Data Type	Optional	Description
<code>templateName</code>	<code>string</code>	No	The absolute pathname of the template to use.
<code>interactive</code>	<code>int</code>	No	Specifies whether FrameMaker displays messages and warnings to the user.

## UserCancel

### Description

Determines whether the user has chosen the Cancel command (ESC) since the current callback function was called.

`UserCancel()` is useful for clients that conduct extensive processing that the user may want to cancel. For example, if your client processes all the documents in a book, it can call `UserCancel()` after it processes each document.

If `UserCancel()` returns `True`, your client can abort the processing.

Returns `True` if the user has executed the Cancel gesture, or `False` if the user has not executed the Cancel gesture.

### Returns

`int`

### Syntax

`UserCancel()`

### Parameters

Parameter name	Data Type	Optional	Description
	<code>Void</code>	No	NULL

## SimpleOpen

### Description

Opens a document or book.

If you call `SimpleOpen()` with `interactive` set to `True`, FrameMaker displays the Open dialog box. It uses the path specified by the session property, `OpenDir`, as the default path.

If a warning or error condition arises, FrameMaker notifies the user. For example, if a document uses fonts that are not available, FrameMaker displays a dialog box that allows the user to cancel the operation or to continue and remap the fonts. If you set `interactive` to `False`, FrameMaker does not display the Open dialog box or other messages and warnings. If it is necessary to modify a file to continue opening it, `SimpleOpen()` aborts the operation without notifying the user, and returns 0.

**Note:** You cannot use `SimpleOpen()` to open filterable files. To open filterable files, use [“Open”](#) on page 755

The method returns the document opened, on success.

If `SimpleOpen()` fails, the method assigns an error code to `FA_errno`. The error codes for `SimpleOpen()` are the same as those for [“Open”](#) on page 755.

### Returns

`Object`

### Syntax

`SimpleOpen(fileName, interactive)`

**Parameters**

Parameter name	Data Type	Optional	Description
fileName	string	No	The absolute pathname of the file to open.
interactive	int	No	Specifies whether the FrameMaker product displays messages and warnings to the user. 1 instructs the FrameMaker product to display messages and warnings.

**UpdateMenus**

**Description**

Updates the top order menu set. For example, when a new menu file is read, or when a view only, book menu, quick menu or similar is loaded, this updates the newly read menus (in case of menu file) or different top menu bar (in case of view only, book and quick menus).

**Returns**

Void

**Syntax**

UpdateMenus ( )

**Parameters**

Parameter name	Data Type	Optional	Description
	Void	No	NULL

**AddMenu**

**Description**

Adds a menu to a specified menu.

The method returns `FE_Success` on success.

On failure, the method assigns one of the following values to `FA_errno`:

Error	Reason
Constants.FE_WrongProduct (-60)	Invalid product.
Constants.FE_BadOperation (-27)	One of: <ul style="list-style-type: none"> <li>• menu is null or empty</li> <li>• tomenu is null or empty</li> <li>• value of menu is already defined as a command</li> </ul>

Error	Reason
Constants.FE_BadParameter (-43)	One of: <ul style="list-style-type: none"> <li>• tomenu and menu have the same values</li> <li>• tomenu is not previously defined</li> <li>• tomenu is not a menu</li> </ul>

**Returns**

int

**Syntax**

AddMenu(toMenu, menu, label)

**Parameters**

Parameter name	Data Type	Optional	Description
toMenu	string	No	The menu tag to which the menu is to be added.
menu	string	No	The new menu which is being added to the existing menu.
label	string	No	The name by which the newly added menu is visible.

**RemoveMenu**

**Description**

Deletes the specified menu.

The method returns FE\_Success on success.

On failure, the method assigns one of the following values to FA\_errno:

Error	Reason
Constants.FE_WrongProduct (-60)	Invalid product.

**Returns**

int

**Syntax**

RemoveMenu(fromMenu, label)

**Parameters**

Parameter name	Data Type	Optional	Description
fromMenu	string	No	The parent menu from which the menu is to be deleted.
label	string	No	The menu to delete.



## MenuExists

### Description

Indicates whether a specified menu exists.

The method returns 1 if the menu exists or 0 if the menu does not exist.

### Returns

int

### Syntax

`MenuExists (menu)`

### Parameters

Parameter name	Data Type	Optional	Description
menu	string	No	The menu to be checked for existence.

## ShortcutExists

### Description

Indicates whether a specified command exists.

The method returns 1 if the command exists or 0 if the command does not exist.

### Returns

int

### Syntax

`ShortcutExists (shortcut)`

### Parameters

Parameter name	Data Type	Optional	Description
shortcut	string	No	The command to be checked for existence.

## GetExportDefaultParams

### Description

Returns the PropVals structure containing the default export parameters.

### Returns

[PropVals](#)

### Syntax

`GetExportDefaultParams ()`

### Parameters

Parameter name	Data Type	Optional	Description
	Void	No	Null

## PrintExportStatus

### Description

TBD

### Returns

Void

### Syntax

PrintExportStatus(p)

### Parameters

Parameter name	Data Type	Optional	Description
p	PropVals	No	TBD

## ErrorName

### Description

Returns the string corresponding to the value of `FA_errno`.

### Returns

string

### Syntax

ErrorName(i)

### Parameters

Parameter name	Data Type	Optional	Description
i	int	No	TBD

## PrintErrno

### Description

Prints the value of `FA_errno` to the console.

### Returns

Void

### Syntax

```
PrintErrno(i)
```

### Parameters

Parameter name	Data Type	Optional	Description
i	int	No	TBD

## NotificationName

### Description

TBD

### Returns

string

### Syntax

```
NotificationName(i)
```

### Parameters

Parameter name	Data Type	Optional	Description
i	int	No	TBD

## AllocatePropVals

### Description

Allocates memory for a property list.

On success, the method returns a property list (a `PropVals` data structure). The returned `PropVals` structure references memory that is allocated by the method. If `AllocatePropVals()` fails, the method sets the `len` field of the returned structure to 0.

### Returns

[PropVals](#)

### Syntax

```
AllocatePropVals(numProps)
```

### Parameters

Parameter name	Data Type	Optional	Description
numProps	int	No	The number of properties in the property list.

## Open

### Description

Opens a document or book. It can also create a new document.

`Open()` allows you to specify a property list telling FrameMaker how to open or create the file and how to deal with error and warning conditions.

For example, you can specify whether to abort or to continue opening a document if it contains fonts that are not available. If the file is already open and invisible, it will make the file visible.

To get a property list to specify for the `openParams` parameter, use `GetOpenDefaultParams()` or create the list from scratch.

To create a new document with `Open()`, set the `Constants.FS_NewDoc (29)` property in the `openParams` property list to `True`.

**Note:** When creating a new document (`Constants.FS_NewDoc (29)` is `True`) and you display the New dialog box (`Constants.FS_ShowBrowser (1)` is `True`) and if the user clicks either Portrait, Custom, or Landscape in the New dialog box, `Open()` does not create a document. Instead, it returns 0 and sets `FA_errno` to either `Constants.FE_WantsPortrait (-48)`, `Constants.FE_WantsCustom (-46)`, or `Constants.FE_WantsLandscape (-47)`, respectively. It is up to your client to create a portrait, custom, or landscape document.

**Note:** For more information on creating custom documents, see “CustomDoc” on page 724.

The method returns the document or book if it opens it successfully, or 0 if an error occurs.

The property list to which `openReturnParams` is set has the properties shown in the following table:

Property	Meaning and possible values
<code>Constants.FS_OpenedFileName (1)</code>	A string that specifies the opened file's pathname. If you scripted <code>Constants.FS_ShowBrowser (1)</code> , or the file was filtered, or you did not specify the pathname, this pathname can be different from the one you specified in the Open script.
<code>Constants.FS_OpenNativeError (2)</code>	The error condition; normally the same value as <code>FA_errno</code> . If the file is opened successfully, it is set to <code>FE_Success</code> .
<code>Constants.FS_OpenStatus (3)</code>	A bit field indicating what happened when the file was opened.

Both the `Constants.FS_OpenNativeError (2)` property and the `FA_errno` global variable indicate the result of a call to `Open()`. The following table lists the possible status flags and the `FA_errno` and `Constants.FS_OpenNativeError (2)` values associated with them.

<b>FS_OpenNativeError and FA_errno values</b>	<b>Possible FS_OpenStatus flags</b>
Constants.FE_Success (0) (file was opened)	<ul style="list-style-type: none"> <li>• Constants.FV_FileHasNewName (32) - Filename was changed from the name specified in the <code>Open ()</code> call.</li> <li>• Constants.FV_RecoverFileUsed (33) - Recover file was present, and it was used.</li> <li>• Constants.FV_AutoSaveFileUsed (34) - Autosave file was present, and the user or the <code>Open ()</code> script chose to use it.</li> <li>• Constants.FV_FileWasFiltered (35) - File was filterable and it was filtered.</li> <li>• Constants.FV_FontsWereMapped (36) - The document contained unavailable fonts, which were mapped to substitute fonts.</li> <li>• Constants.FV_FontMetricsChanged (37) - The file contained fonts with changed metrics, but it was opened anyway.</li> <li>• Constants.FV_FontsMappedInCatalog (38) - The Paragraph or Character Catalog used unavailable fonts, which were mapped to substitute fonts.</li> <li>• Constants.FV_LanguagesWerentFound (39) - The document used some unavailable languages, but it was opened anyway.</li> <li>• Constants.FV_BeefyDoc (40) - The document file was extremely large, but it was opened anyway.</li> <li>• Constants.FV_FilesOldVersion (41) - The file was from an old FrameMaker product version, but the user or the <code>Open ()</code> script chose to open it anyway.</li> <li>• Constants.FV_FileStructureStripped (42) - The file had FrameMaker features, which the user or the <code>Open ()</code> script chose to strip.</li> <li>• Constants.FV_FilesText (43) - The file was a Text Only file, but the user or the <code>Open ()</code> script chose to open it anyway.</li> <li>• Constants.FV_OpenedViewOnly (44) - The user or the <code>Open ()</code> script chose to open the file as a View Only file.</li> <li>• Constants.FV_EditableCopyOpened(45) - The file was in use and the user or the <code>Open ()</code> script opened an editable copy.</li> <li>• Constants.FV_BadFileRefsWereMapped (46) - File reference contained illegal characters; the illegal characters were converted to something safe.</li> <li>• Constants.FV_ReferencedFilesWerentFound (47) - Imported graphics files could not be found, but the file was opened anyway.</li> </ul>
Constants.FE_Success (0) (file was opened)	<ul style="list-style-type: none"> <li>• Constants.FV_FileAlreadyOpen (48) - The file was in use and the user or the <code>Open ()</code> script opened another copy.</li> <li>• Constants.FV_UnresolvedXRefs (49) - There were unresolved cross-references, but the file was opened anyway.</li> <li>• Constants.FV_UnresolvedTextInsets(50) - There were unresolved text insets, but the file was opened anyway.</li> </ul>
Constants.FE_SystemError (-42) (file was not opened)	<ul style="list-style-type: none"> <li>• Constants.FV_TooManyWindows(128) - Too many windows were open.</li> <li>• Constants.FV_BadTemplate (129) - A bad template was specified.</li> <li>• Constants.FV_FileNotReadable (130) - Do not have read permission for the file.</li> </ul>

FS_OpenNativeError and FA_errno values	Possible FS_OpenStatus flags
Constants.FE_BadParameter (-43) (file was not opened)	<ul style="list-style-type: none"><li>• Constants.FV_FileHadStructure(64) - File had FrameMaker features, but current FrameMaker product is not FrameMaker.</li><li>• Constants.FV_FileAlreadyOpenThisSession (65) - File is already open and script disallowed opening another copy.</li><li>• Constants.FV_BadFileType (66) - File was an executable file or other unreadable type.</li><li>• Constants.FV_BadFileName (67) - Specified filename was invalid.</li><li>• Constants.FV_CantNewBooks (68) - Script specified a book that did not exist (the Open operation cannot create a new book).</li><li>• Constants.FV_BadScriptValue (70) - <code>Open()</code> script contained an invalid property value.</li><li>• Constants.FV_MissingScript (71) - <code>Open()</code> was called without a script.</li><li>• Constants.FV_CantForceOpenAsText (72) - <code>Open()</code> script attempted to open the file as text, but file was of the wrong type.</li><li>• Constants.FV_DisallowedType (73) - File was a Frame binary document and the <code>Open()</code> script disallowed it.</li><li>• Constants.FV_DocDamagedByTextFilter (74) - File was a text document and was damaged when it was filtered.</li><li>• Constants.FV_DocHeadersDamaged (75) - The document headers were damaged (probably because of a file system problem).</li><li>• Constants.FV_DocWrongSize (76) - File is the wrong size (probably because of a file system problem).</li><li>• Constants.FV_ChecksumDamage (77) - Bad checksum.</li></ul>

FS_OpenNativeError and FA_errno values	Possible FS_OpenStatus flags
<p>Constants.FE_Canceled (-44) (file was not opened)</p>	<ul style="list-style-type: none"> <li>• Constants.FV_CancelUseRecoverFile (96) - A recover file was present, so the user or the <code>Open()</code> script canceled the Open operation.</li> <li>• Constants.FV_CancelUseAutoSaveFile (97) - An Autosave file was present, so the user or the <code>Open()</code> script canceled the Open operation.</li> <li>• Constants.FV_CancelFileIsText (98) - The file was text, so the user or the <code>Open()</code> script canceled the Open operation.</li> <li>• Constants.FV_CancelFileIsInUse (99) - The file was in use, so the user or the <code>Open()</code> script canceled the Open operation.</li> <li>• Constants.FV_CancelFileHasStructure (100) - The file had structure, so the user or the script canceled the Open operation.</li> <li>• Constants.FV_CancelReferencedFilesNotFound (101) - The file contained referenced files that were not available, so the user or the <code>Open()</code> script canceled the Open operation.</li> <li>• Constants.FV_CancelLanguagesNotFound (102) - The file contained languages that were not available, so the user or the <code>Open()</code> script canceled the Open operation.</li> <li>• Constants.FV_CancelFontsMapped (103) - The document contained fonts that needed to be mapped to other fonts, so the user or the <code>Open()</code> script canceled the Open operation.</li> <li>• Constants.FV_CancelFontMetricsChanged (104) - The file contained fonts with changed metrics, so the user or the <code>Open()</code> script canceled the Open operation.</li> <li>• Constants.FV_CancelFontsMappedInCatalog (105) - The document's Character Catalog or Paragraph Catalog contained fonts that needed to be mapped to other fonts, so the user or the <code>Open()</code> script canceled the Open operation.</li> <li>• Constants.FV_CancelFileIsDoc (106) - The file was a document and the <code>Open()</code> script disallowed it.</li> <li>• Constants.FV_CancelFileIsMIF (107) - The file was a MIF file and the <code>Open()</code> script disallowed it.</li> <li>• Constants.FV_CancelBook (108) - The file was a book and the <code>Open()</code> script disallowed it.</li> <li>• Constants.FV_CancelBookMIF (109) - The file was a MIF file and the <code>Open()</code> script disallowed it.</li> <li>• Constants.FV_CancelFileIsFilterable (110) - The file was a filterable file and the <code>Open()</code> script disallowed it.</li> <li>• Constants.FV_CancelFileIsOldVersion (111) - The file was from an old version of a FrameMaker product, so the user or the <code>Open()</code> script canceled the Open operation.</li> <li>• Constants.FV_UserCanceled (112) - The user canceled the Open operation.</li> <li>• Constants.FV_CancelFileBrowser (113) - The user canceled the Open operation from the file browser.</li> <li>• Constants.FV_CancelBeefyDoc (114) - The document was very large and the user or the script disallowed it.</li> <li>• Constants.FV_CancelFileIsSgml (115) - The file was an SGML document and the <code>Open()</code> script disallowed it.</li> <li>• Constants.FV_CancelTempDiskFull (118) - There was insufficient room on the disk to cache data while opening the file.</li> <li>• Constants.FV_CancelFileIsXml (119) - The file was an XML document and the <code>Open()</code> script disallowed it.</li> </ul>

FS_OpenNativeError and FA_errno values	Possible FS_OpenStatus flags
<ul style="list-style-type: none"> <li>Constants.FE_Success (0)</li> <li>Constants.FE_Canceled (-44)</li> <li>Constants.FE_FailedState (-45)</li> <li>Constants.FE_CanceledByClient (-86)</li> </ul>	<ul style="list-style-type: none"> <li>Constants.FV_LockWasReset (0) - File lock was reset.</li> <li>Constants.FV_LockNotReset (1) - File had a lock that was not reset.</li> <li>Constants.FV_LockCouldntBeReset (2) - File had a lock that could not be reset.</li> <li>Constants.FV_FileWasInUse (3) - File was in use.</li> <li>Constants.FV_FilesViewOnly (4) - File is a View Only file.</li> <li>Constants.FV_LockWasInvalid (5) - File had an invalid lock.</li> <li>Constants.FV_FilesNotWritable (6) - The file was not writable, and the user canceled the open through the alert.</li> <li>Constants.FV_FileModDateChanged (7) - The file has changed since the last time it was opened or saved in the current session.</li> </ul>

To determine if a particular FS\_OpenStatus bit is set, use [“CheckStatus”](#) on page 721.

**Returns**

Object

**Syntax**

Open(fileName, openParams, openReturnParams)

**Parameters**

Parameter name	Data Type	Optional	Description
fileName	string	No	The full pathname of the file to open. If you are using Open() to create a document, specify the name of the template to use.
openParams	PropVals	No	A property list telling the FrameMaker product how to open the file and how to respond to errors and other conditions. To use the default list, specify NULL.
openReturnParams	PropVals	No	A property list that returns the filename and provides information about how FrameMaker opened the file. It must be initialized before you call Open().

**ScrollBox**

**Description**

Displays an array of items and allows the user to choose one.

*Note: If you set default to -1, always check to make sure the value returned in selected\_item is 0 or greater before you use it as an array index. If you set default to -1 and the user clicks OK without choosing an item, the value returned in selected\_item will be -1.*

Returns 0 if the user clicked OK, or a nonzero value if the user clicked Cancel or an error occurred.

On failure, the method assigns the following value to FA\_errno:



**Function Summary**

Error	Reason
Constants.FE_Transport (-1)	The user clicked Cancel, or a transport error occurred.

**Returns**

int

**Syntax**

```
ScrollBox(title, stringslist, _default)
```

**Parameters**

Parameter name	Data Type	Optional	Description
title	string	No	The title that appears on the dialog box.
stringslist	Strings	No	The list of items to appear in the scroll list.
_default	int	No	The index of the item that is selected when the dialog box first appears. For no default, specify -1.

**ChooseFile****Description**

Displays dialog boxes similar to a FrameMaker's Open and Save dialog boxes. The method displays directories and files in a scroll list and allows the user to choose a file or directory.

You can set the following values for `mode`:

Mode Constant	Dialog Box Type
Constants.FV_ChooseSelect (0)	Dialog box that allows the user to choose a file by clicking Select. It provides an input field into which the user can type a filename.
Constants.FV_ChooseOpen (1)	Dialog box that allows the user to choose a file by clicking Open. It provides an input field into which the user can type a filename.
Constants.FV_ChooseSave (2)	Dialog box that allows the user to save a file. It provides Save and Cancel buttons and an input field.
Constants.FV_ChooseOpenDir (3)	Dialog box that allows the user to choose a directory.

The method returns 0 if the user clicked Open, Select, Use, or Save; a nonzero value if the user clicked Cancel or an error occurred.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

**Returns**

int

**Syntax**

```
ChooseFile(title, directory, stuffVal, mode)
```

**Parameters**

Parameter name	Data Type	Optional	Description
title	string	No	The message that appears in the dialog box.
directory	string	No	The default directory when the dialog box is first displayed. If you specify an empty string, the last directory used by an ESTK client is used. If no ESTK client has used a directory, the directory specified by the session property, <code>OpenDir</code> , is used.
stuffVal	string	No	The default value that appears in the input field when the dialog box first appears. If the dialog box type specified by mode does not have an input field, this string is ignored.
mode	int	No	A constant specifying the type of dialog box.

**PromptInt**

**Description**

Displays a dialog box that prompts the user for a single integer value. It allows you to provide a default value, which appears in the entry field when the dialog box appears. The dialog box has OK and Cancel buttons.

`PromptInt()` does not assign a value to `*intp` if the user clicks Cancel. If the user types alphabetic text after a number, the method ignores the text and just returns the value of the number.

The method returns 0 if the user clicked OK; a nonzero value if the user clicked Cancel or an error occurred.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

**Returns**

int

**Syntax**

`PromptInt(message, stuffVal)`

**Parameters**

Parameter name	Data Type	Optional	Description
message	string	No	The message that appears in the dialog box. It must be 255 characters or less.
stuffVal	string	No	The default value that appears in the input field when the dialog box is first displayed.

**PromptMetric**

**Description**

Displays a dialog box that prompts the user for a single metric value. It allows you to provide a default value, which appears in the entry field when the dialog box appears. The dialog box has OK and Cancel buttons.

**Function Summary**

`PromptMetric()` does not assign a value to `*metricp` if the user clicks Cancel.

`PromptMetric()` dialog boxes behave like metric dialog boxes in the user interface. If the user types a number followed by a string that represents a unit (for example 10pts or 5"), the method converts the number into the equivalent number of metric units. If the user does not specify a unit, the method uses points (metric 65536).

The method returns 0 if the user clicked OK; a nonzero value if the user clicked Cancel or an error occurred.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

**Returns**

int

**Syntax**

```
PromptMetric(message, stuffVal, defaultunit)
```

**Parameters**

Parameter name	Data Type	Optional	Description
message	string	No	The message that appears in the dialog box. It must be 255 characters or less.
stuffVal	string	No	The default value that appears in the input field when the dialog box is first displayed.
defaultunit	int	No	The metric unit to use if the user does not specify one.

**PromptString****Description**

Displays a dialog box that prompts the user for a single string value. It allows you to provide a default string, which appears in the entry field when the dialog box appears. The dialog box has OK and Cancel buttons.

**Note:** Do not call `PromptString()` to prompt the user for a pathname. If the user enters a pathname as a string, the backslash character (\) is interpreted as a special escape character. For example, the characters `\s` represent a space. If the user enters the pathname `c:\sample`, this string is interpreted as `c: \sample`. To prompt the user for a pathname, use ["ChooseFile"](#) on page 760 to display a file selection dialog box.

The method returns 0 if the user clicked OK; a nonzero value if the user clicked Cancel or an error occurred.

On failure, the method assigns the following value to `FA_errno`:

Error	Reason
Constants.FE_Transport (-1)	A transport error occurred.

**Returns**

string

### Syntax

PromptString(message, stuffVal)

### Parameters

Parameter name	Data Type	Optional	Description
message	string	No	The message that appears in the dialog box. It must be 255 characters or less. Newline and linefeed characters are ignored.
stuffVal	string	No	The default value that appears in the input field when the dialog box is first displayed.

## Service

### Description

Services calls from FrameMaker to the ESTK API.

`Service()` waits for and dispatches a single Windows message. If your application contains its own message processing loop you need not call this routine.

It is not feasible for console applications to receive ESTK notifications since there is no way for them to both wait for user input and call `Service()`. Consequently, console applications should not register for notifications.

The method returns the number of bits the call to `select()` selected, if any.

### Returns

int

### Syntax

`Service()`

## FileScrollBox

### Description

Displays a file scrollbox using the given title and the given directory as the path.

### Returns

int

### Syntax

FileScrollBox(title, directory)

### Parameters

Parameter name	Data Type	Optional	Description
title	string	No	The title to display on the scroll box.
directory	string	No	The directory path to display.



# Chapter 7: CMS Connector Framework

## CMS API Data Structures and Enum Constants

### CMSResult

#### CMSResult

##### Description

Specifies the state of a Command's result for CMSCommand.

##### Returns

CMSResult

##### Syntax

```
CMSResult (status, opResult, message, cmsItems);
```

##### Parameters

Parameter name	Data Type	Optional	Description
status	int	No	Command's status
opResult	int	No	Operation's result. If CMS Command needs CMSTree update, assign CMSOpResult values (See following enum),else can return any value depending on operation. For example:  opResult = True/False for FA_CMSIsValidCommand ,  opResult = CMSPropertyNewMaxOpCode for FA_CMSGetPropertyMaxOpCode
message	string	No	If operation fails, user can send error message to FrameMaker.For FA_CMSObjectOpenReadOnly, FA_CMSObjectEdit command, user can return file-name which is downloaded.
cmsItems	CMSObjects	No	List of CMS object

The possible values of the CMSResult.opResult field are:

opResult constant	Meaning
Constants.CMSOpNone	None
Constants.CMSOpItemUpdated	CMS item is updated
Constants.CMSOpDependentsUpdated	Dependents are updated
Constants.CMSOpDependentsDeleted	Dependents are deleted
Constants.CMSOpItemDeleted	CMS item is deleted
Constants.CMSOpChildAdded	Child is added

Constants.CMSOpRootUpdated	Root is updated
----------------------------	-----------------

If CMS SetProperty() fails, the API assigns following values to FA\_errno:

Error	Reason
Constants.FE_CMSBadSessionId	The client specified an invalid session ID.
Constants.FE_CMSBadObjectId	The client specified an invalid cms object ID.
Constants.FE_CMSRootObjectExists	The API tries to set a root which already exists.
Constants.FE_CMSBadItemType	The file type expected by the cms object does not match the valid file type.
Constants.FE_CMSBadItemFiletype	The item type expected by the cms object does not match the valid item type
Constants.FE_CMSBadItemContainerType	The container value expected by the cms object is not properly set

## CMSItemFileType

### Description

Enum constants used to determine File-Type of a CMS Object.

The possible values of the Constants.FP\_CMSItemProperty\_ItemFileType fields are:

File Type constant	Meaning
Constants.FV_CMSItemFileTypeValue_Xml	File type is XML
Constants.FV_CMSItemFileTypeValue_FmDoc	File type is FM
Constants.FV_CMSItemFileTypeValue_Mif	File type is MIF
Constants.FV_CMSItemFileTypeValue_DitaMap	File type is DITAMAP
Constants.FV_CMSItemFileTypeValue_DitaTopic	File type is DITA topic
Constants.FV_CMSItemFileTypeValue_FmBook	File type is Book
Constants.FV_CMSItemFileTypeValue_Text	File type is text
Constants.FV_CMSItemFileTypeValue_Img	File type is image
Constants.FV_CMSItemFileTypeValue_General	File type is General

## CMSProperties

### Description

A CMSProperties structure specifies a set of CMS object properties.

## CMSMenuItem

### Description

The CMSMenuItem structure describes a custom menu definition.

This structure is used for creating a custom menu entry in the context menu available in CMS tree and CMS dialogs.

## CMSCheckinParam

### Description

The `CMSCheckinParam` structure describes the checkin parameter.

This structure is returned by `CMSShowCheckinUI` API for getting all the user interface state after user accepts the dialog changes by pressing OK button.

## CMSDeleteParam

### Description

The `CMSDeleteParam` structure describes the delete parameter.

This structure is returned by `CMSShowDeleteUI` API for getting all the user interface state after the user accepts the dialog changes by clicking OK.

## CMSInfo

### Description

The `CMSInfo` structure describes a single CMS registration information definition

## CMSInfos

### Description

The `CMSInfos` structure specifies a set of CMS registration information.

## Error Codes

If the CMS API encounters an error condition, the API assigns an error code to the global variable `FA_erno`. The following table lists the error codes and their meanings. Error codes are also listed in the `fcmsapi.h` header file.

Error code	Meaning
<code>Constants.FE_CMSNameAlreadyRegistered</code>	The API attempted to register a CMS that is already registered.
<code>Constants.FE_CMSBadSessionId</code>	The client specified an invalid session ID.
<code>Constants.FE_CMSBadObjectId</code>	The client specified an invalid CMS object ID.
<code>Constants.FE_CMSSessionFailed</code>	The client failed to create a session.
<code>Constants.CMSBadCommandId</code>	The client specified an invalid command ID.
<code>Constants.FE_CMSObjectCreationFailed</code>	The <code>CMSCreateObject</code> API fails to create a CMS object.
<code>Constants.FE_CMSRootObjectExists</code>	The API tried to set a root that already exists.
<code>Constants.FE_CMSBadItemFileType</code>	The file type expected by the CMS object does not match the valid file type.



Constants.FE_CMSBadItemType	The item type expected by the CMS object does not match the valid item type.
Constants.FE_CMSBadItemContainerType	The container value expected by the cms object is not properly set
Constants.FE_CMSSessionCreationFailed	If Session creation fails, set status to this value.
Constants.FE_CMSIsValidCMSCommand	If user wants FrameMaker to take care of IsValidCMSCommand, set opResult to this value.
Constants.FE_CMSFailedLogin	The CMSLogin API fails to log into a CMS.
Constants.FE_CMSFailedLogout	The CMSLogout API fails to log out from a CMS.
Constants.FE_CMSFailedCheckout	The CMSCheckout API failed to checkout a file.
Constants.FE_CMSFailedCheckin	The CMSCheckin API failed to check in a file
Constants.FE_CMSFailedCancelCheckout	The CMSCancelCheckout API fails to cancelcheckout of a file.
Constants.FE_CMSFailedDelete	The CMSDelete API failed to delete a cms object
Constants.FE_CMSFailedOpenFile	The CMSOpenFile API failed to open a file
Constants.FE_CMSFailedUploadObject	The CMSUploadObject API failed to upload afile or folder.
Constants.FE_CMSFailedDownloadObject	The CMSzDownloadObject API failed to download a file.
Constants.FE_CMSFailedGetItemFrompath	The GetCMSObjectFromPath API failed to return a CMS object from server path.

## CMS API functions

A list of the prominent CMS API functions:

- [“CMSRegister”](#) on page 722
- [“CMSCreateObject”](#) on page 512
- [“CMSEnableCommand\(\)”](#) on page 501
- [“CMSDisableCommand\(\)”](#) on page 502
- [“CMSAddMenuEntry”](#) on page 723
- [“CMSGetCmsIdFromName”](#) on page 723
- [“CMSGetCMSInfo\(\)”](#) on page 508
- [“CMSGetCmsIdFromSession”](#) on page 512
- [“CMSConfigLoginUI\(\)”](#) on page 509
- [“CMSShowCheckoutUI\(\)”](#) on page 502
- [“CMSShowCheckinUI\(\)”](#) on page 503
- [“CMSShowCancelCheckoutUI\(\)”](#) on page 504
- [“CMSShowDeleteUI\(\)”](#) on page 505
- [“CMSShowCommonListUI\(\)”](#) on page 505

[“CMSShowPropertyUI\(\)”](#) on page 506

[“CMSShowPropertyUIWithTitle”](#) on page 507

[“CMSGetProperty\(\)”](#) on page 500

[“CMSGetProperties\(\)”](#) on page 501

[“CMS SetProperty\(\)”](#) on page 499

[“CMSSetProperties\(\)”](#) on page 499

## APIs to automate CMS connector functionality

### CMSLogin

#### Description

Logs into a particular CMS based on the connection details

#### Returns

Returns the handle of the new CMS connection if the operation is successful. Else sets FA\_errno to Constants.FE\_CMSErrordLogin.

#### Syntax

```
CMSLogin (setVal);
```

#### Parameters

Parameter name	Data Type	Optional	Description
setVal	IdValuePairs	No	Id value pairs to specify the connection parameter. The valid Ids are: Constants.FV_CMSErrordNameId - Name of the connection Constants.FV_CMSErrordConnTypeId - Connection Type Constants.FV_CMSErrordServerId - Server Name Constants.FV_CMSErrordUserNameId - User Name Constants.FV_CMSErrordPasswordId - Password Constants.FV_CMSErrordUserField1 - Optional User Field1 Constants.FV_CMSErrordRepositoryId - Repository name for documentum Constants.FV_CMSErrordUserField2 - Optional User Field2

### CMSLogout

#### Description

Logs out the user from a particular CMS connection

**Returns**

Returns `FE_Success` if the operation is successful, else sets `FA_errno` to `Constants.FE_CMSFailedLogout`

**Syntax**

`CMSLogout (cmsSessionId);`

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS session

## CMSCheckout

**Description**

Checks out a file from the CMS

**Returns**

Returns `FE_Success` if the operation is successful. Else sets `FA_errno` to `Constants.FE_CMSFailedCheckout`

**Syntax**

`CMSCheckout (cmsSessionId, cmsObjectId, rootWithDescendants);`

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
rootWithDescendants	bool	No	True if checked out root with descendants

## CMSCheckin

**Description**

Checks in a file into the CMS

**Returns**

Returns `FE_Success` if the operation is successful, else sets `FA_errno` to `Constants.FE_CMSFailedCheckin`

**Syntax**

`CMSCheckin (cmsSessionId,objectId, checkinParam);`

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
checkinParam	IdValuePairs	No	Id value pairs to specify the checkin parameter. The valid Ids are: Constants.FV_CMSCheckinMakeCurrentVersionId Constants.FV_CMSCheckinKeepLocalCopyId Constants.FV_CMSCheckinMinorVersionId Constants.FV_CMSCheckinVersionLabelId Constants.FV_CMSCheckinDescriptionId Constants.FV_CMSCheckinCommentId

**CMSCancelCheckout**

**Description**

Cancels check out of a file from the CMS

**Returns**

Returns FE\_Success if the operation is successful, else sets FA\_errno to Constants.FE\_CMSCancelCheckoutFailed

**Syntax**

CMSCancelCheckout (cmsSessionId, objectId);

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object

**CMSDelete**

**Description**

Deletes a file or a folder from CMS

**Returns**

Returns FE\_Success if the operation is successful, else sets FA\_errno to Constants.FE\_CMSCancelDeleteFailed

**Syntax**

CMSDelete (cmsSessionId, cmsObjectId, deleteParams);

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
deleteParams	IdValuePairs	No	Id value pairs to specify the delete parameter. The valid Ids are: Constants.FV_CMSCCommandDeleteAllVersionId Constants.FV_CMSCCommandDeleteAllDependentsId

**CMSOpenFile**

**Description**

Opens a file or a book from CMS in FrameMaker

**Returns**

Returns the handle of the file or book if the operation is successful. Else sets FA\_errno to Constants.FE\_CMSErrorFailedOpenFile

**Syntax**

CMSOpenFile (cmsSessionId, cmsObjectId, openParams);

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
openParams	IdValuePairs	No	Id value pairs to specify the open parameter. The valid Ids are: Constants.FV_CMSCCommandOpenReadOnlyId Constants.FV_CMSCCommandSilentOpenId

**CMSUploadObject**

**Description**

Uploads a file or a folder into the CMS

**Returns**

Returns FE\_Success if the operation is successful. Else sets FA\_errno to Constants.FE\_CMSErrorFailedUploadObject

**Syntax**

CMSUploadObject (cmsSessionId, cmsObjectId, localFilePath);

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object
localFilePath	string	No	The full pathname of the file or folder to upload

**CMSDownloadObject**

**Description**

Downloads a file from the CMS

**Returns**

Returns the local file path of the downloaded file if the operation is successful. Else sets FA\_errno to Constants.FE\_CMSFailedDownloadObject

**Syntax**

`CMSDownloadObject (cmsSessionId, cmsObjectId);`

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
cmsObjectId	int	No	The ID of the CMS Object

**GetCMSObjectFromPath**

**Description**

Gets CMS object from a URL path

**Returns**

Returns the handle of a CMS object if the operation is successful. Else sets FA\_errno to Constants.FE\_CMSFailedGetItemFrompath

**Syntax**

`GetCMSObjectFromPath (cmsSessionId,urlPath);`

**Parameters**

Parameter name	Data Type	Optional	Description
cmsSessionId	int	No	The ID of the CMS Session
urlPath	string	No	The url pathname of the file or folder