

What's new in InDesign CS5 Scripting DOM

1 | New Object Members

The data presented below are extracted from the Scripting DOM reference. They only include the **objects** and **object members** introduced in InDesign 7.0. A full hypertext reference is available here: <http://www.jongware.com/idjshelp.html>

This document has been automatically generated by parsing the **Type Library** files provided by Adobe from InDesign 3.0 (CS) to InDesign 7.0 (CS5). It may contain typos or adventitious case mistakes due to the automation process.

To limit the size of this document we didn't include the generic new members that concern almost all UI objects in InDesign CS5: **events**, **eventListeners**, **addEventListener()**, and **removeEventListener()**.

- To easily browse the contents by alphabetic order, use the PDF bookmarks: View > Navigation Panels > Bookmarks.
- A separate document focuses on new Enumeration values.

LEGEND:

| | |
|--------|---|
| Type | Primitive argument/value type (string, boolean, etc.) |
| {Type} | Any DOM object type. |
| #Type | Enumeration type. |
| • | Indicates a read-only property. |

Application

Object available in ID 3, 4, 5, 6, 7

| APPLICATION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|--|--|
| applyShortcutSet | <code>void applyShortcutSet(variant Name)</code> | Applies the specified shortcut set file. No string will apply the default shortcut set. / Name: The shortcut set. Type: String. |
| cancelAllTasks | <code>void cancelAllTasks()</code> | Cancels all the background tasks. |
| changeTransliterate | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| clearFrameFittingOptions | <code>void clearFrameFittingOptions()</code> | Removes the frame fittings options and resets it to the initial state. |
| findTransliterate | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| loadMotionPreset | <code>{MotionPreset} loadMotionPreset(string From)</code> | Load motion preset from the specified file. / Return value: The motion preset that is imported. / From: The Flash motion preset file. |
| setApplicationPreferences | <code>void setApplicationPreferences(variant ApplicationPreferences)</code> | Sets the application's preferences. / ApplicationPreferences: The IDML defaults file or enumeration. Type: File (String) or idLanguageAndRegion enumerator. |
| waitForAllTasks | <code>variant waitForAllTasks()</code> | Waits for all the background tasks to finish. / Return value: A list of task states for task that finished. Type: Array of idTaskState enumerators. |
| backgroundTasks | <code>{BackgroundTasks} *backgroundTasks</code> | A collection of background task objects. |
| changeTransliteratePreferences | <code>variant changeTransliteratePreferences</code> | Change transliterate preferences. Type: ChangeTransliteratePreference or idNothingEnum enumerator. |
| cjkGridPreferences | <code>{CjkGridPreference} *cjkGridPreferences</code> | CJK grid preference settings. |
| compositeFonts | <code>{CompositeFonts} *compositeFonts</code> | A collection of composite fonts. |
| findChangeTransliterateOptions | <code>variant findChangeTransliterateOptions</code> | Find/change transliterate options. Type: FindChangeTransliterateOption or idNothingEnum enumerator. |
| findTransliteratePreferences | <code>variant findTransliteratePreferences</code> | Find transliterate preferences. Type: FindTransliteratePreference or idNothingEnum enumerator. |
| gridPrintingPreferences | <code>{GridPrintingPreference} *gridPrintingPreferences</code> | Grid printing preference and export settings. |
| idleTasks | <code>{IdleTasks} *idleTasks</code> | All attachable idle tasks. |
| interactivePDFExportPreferences | <code>{InteractivePDFExportPreference} *interactivePDFExportPreferences</code> | Interactive PDF export preferences. |
| kinsokuTables | <code>{KinsokuTables} *kinsokuTables</code> | A collection of kinsoku tables. |
| layoutGridData | <code>{LayoutGridDataInformation} *layoutGridData</code> | Default properties specific to layout grids. |
| liveScreenDrawing | <code>#LiveDrawingOptions liveScreenDrawing</code> | Delay before mouse operations trigger live screen drawing of page items. |
| mojikumiTables | <code>{MojikumiTables} *mojikumiTables</code> | A collection of mojikumi tables. |
| mojikumiUIPreferences | <code>{MojikumiUiPreference} *mojikumiUIPreferences</code> | Mojikumi user interface preference settings. |
| motionPresets | <code>{MotionPresets} *motionPresets</code> | A collection of Motion presets. |
| namedGrids | <code>{NamedGrids} *namedGrids</code> | A collection of named grids. |
| serialNumber | <code>string *serialNumber</code> | The user's serial number. |

| APPLICATION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|--|--|
| <code>storyGridData</code> | <code>{StoryGridDataInformation} *storyGridData</code> | <i>Default properties specific to frame grids.</i> |
| <code>toolBoxTools</code> | <code>{ToolBox} *toolBoxTools</code> | <i>The current tool box states.</i> |
| <code>trackChangesPreferences</code> | <code>{TrackChangesPreference} *trackChangesPreferences</code> | <i>Track changes preference settings.</i> |
| <code>watermarkPreferences</code> | <code>{WatermarkPreference} *watermarkPreferences</code> | <i>Watermark preferences.</i> |

Book

Object available in ID 3, 4, 5, 6, 7

| BOOK CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|---|---|
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |

BookContent

Object available in ID 3, 4, 5, 6, 7

| BOOKCONTENT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------|---------------------------|-------------------------------------|
| <code>name</code> | <code>string *name</code> | <i>The name of the BookContent.</i> |

Button

Object available in ID 3, 4, 5, 6, 7

| BUTTON CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>Asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| <code>activeStateIndex</code> | <code>integer activeStateIndex</code> | <i>The index of the active state in the object's states collection.</i> |
| <code>animationBehaviors</code> | <code>{AnimationBehaviors} *animationBehaviors</code> | <i>A collection of animation behaviors.</i> |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | <i>The page item animation settings.</i> |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String).</i> |

| BUTTON CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|---|
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| <code>gotoNextStateBehaviors</code> | <code>{GotoNextStateBehaviors} *gotoNextStateBehaviors</code> | A collection of goto next state behaviors. |
| <code>gotoPreviousStateBehaviors</code> | <code>{GotoPreviousStateBehaviors} *gotoPreviousStateBehaviors</code> | A collection of goto previous state behaviors. |
| <code>gotoStateBehaviors</code> | <code>{GotoStateBehaviors} *gotoStateBehaviors</code> | A collection of goto state behaviors. |
| <code>parentPage</code> | <code>{Page} *parentPage</code> | The page on which this page item appears. |
| <code>paths</code> | <code>{Paths} *paths</code> | A collection of paths. |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| <code>timingSettings</code> | <code>{TimingSetting} *timingSettings</code> | The object timing settings. |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String). |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | The shape to apply to the top right corner of rectangular shapes. |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String). |
| <code>visible</code> | <code>boolean visible</code> | If true, the Button is visible. |

Cell

Object available in ID 3, 4, 5, 6, 7

| CELL CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|--|
| <code>changeTransliterate</code> | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| <code>findTransliterate</code> | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| <code>multiStateObjects</code> | <code>{MultiStateObjects} *multiStateObjects</code> | A collection of multi-state objects. |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| <code>writingDirection</code> | <code>#HorizontalOrVertical writingDirection</code> | The direction of the text in the cell. |

Character

Object available in ID 3, 4, 5, 6, 7

| CHARACTER CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|--|
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| changeTransliterate | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| findHyperlinks | <code>{Objects} findHyperlinks(variant SortOrder)</code> | Finds hyperlink sources that intersecting with specified text range. / Return value: The found hyperlinks. / SortOrder: The sort order of found ranges. Type: idRangeSortOrder enumerator. |
| findTransliterate | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| showText | <code>void showText()</code> | Jump to the text range. |
| allowArbitraryHyphenation | <code>boolean allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. |
| autoTcy | <code>const autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. |
| autoTcyIncludeRoman | <code>boolean autoTcyIncludeRoman</code> | If true, auto tcy includes Roman characters. |
| bunriKinshi | <code>boolean bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. |
| characterAlignment | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| characterRotation | <code>double characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. |
| cjkGridTracking | <code>boolean cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. |
| epsTexts | <code>{EPSTexts} •epsTexts</code> | EPSTexts. |
| glyphForm | <code>#AlternateGlyphForms glyphForm</code> | The glyph variant to substitute for standard glyphs. |
| gridAlignment | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| gridGyoudori | <code>const gridGyoudori</code> | The manual gyoudori setting. |
| jidori | <code>const jidori</code> | The number of grid squares in which to arrange the text. |
| keepWithPrevious | <code>boolean keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. |
| kntenAlignment | <code>#KntenAlignment kntenAlignment</code> | The alignment of knten characters relative to the parent characters. |
| kntenCharacterSet | <code>#KntenCharacterSet kntenCharacterSet</code> | The character set used for the custom knten character. Note: Valid only when knten kind is custom. |
| kntenCustomCharacter | <code>string kntenCustomCharacter</code> | The character used for knten. Note: Valid only when knten kind is custom. |

| CHARACTER CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|---|---|
| kntenFillColor | <code>variant</code> kntenFillColor | The swatch (color, gradient, tint, or mixed ink) applied to the fill of knten characters. Type: Swatch or String. |
| kntenFont | <code>variant</code> kntenFont | The font to use for knten characters. Type: Font or String. |
| kntenFontSize | <code>double</code> kntenFontSize | The size (in points) of knten characters. |
| kntenFontStyle | <code>variant</code> kntenFontStyle | The font style of knten characters. Type: String or idNothingEnum enumerator. |
| kntenKind | <code>#KetenCharacter</code> ktenKind | The style of knten characters. |
| kntenOverprintFill | <code>#AdornmentOverprint</code> kntenOverprintFill | The method of overprinting the knten fill. |
| kntenOverprintStroke | <code>#AdornmentOverprint</code> kntenOverprintStroke | The method of overprinting the knten stroke. |
| kntenPlacement | <code>double</code> knenPlacement | The distance between knten characters and their parent characters. |
| kntenPosition | <code>#RubyKetenPosition</code> ktenPosition | The kten position relative to the parent character. |
| kntenStrokeColor | <code>variant</code> knenStrokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of knten characters. Type: Swatch or String. |
| kntenStrokeTint | <code>double</code> knenStrokeTint | The stroke tint (as a percentage) of knten characters. (Range: 0 to 100). |
| kntenTint | <code>double</code> knenTint | The fill tint (as a percentage) of knten characters. (Range: 0 to 100). |
| kntenWeight | <code>double</code> knenWeight | The stroke weight (in points) of knten characters. |
| kntenXScale | <code>double</code> knenXScale | The horizontal size of knten characters as a percent of the original size. |
| kntenYScale | <code>double</code> knenYScale | The vertical size of knen charachers as a percent of the original size. |
| kinsokuHangType | <code>#KinsokuHangTypes</code> kinsokuHangType | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. |
| kinsokuSet | <code>variant</code> kinsokuSet | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator or String. |
| kinsokuType | <code>#KinsokuType</code> kinsokuType | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. |
| leadingAki | <code>double</code> leadingAki | The amount of space before each character. |
| leadingModel | <code>#LeadingModel</code> leadingModel | The point from which leading is measured from line to line. |
| mojikumi | <code>variant</code> mojikumi | The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String or idMojikumiTableDefaults enumerator. |
| multiStateObjects | <code>{MultiStateObjects}</code> *multiStateObjects | A collection of multi-state objects. |
| otfHVKana | <code>boolean</code> otfHVKana | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. |
| otfProportionalMetrics | <code>boolean</code> otfProportionalMetrics | If true, kerns according to proportional CJK metrics in OpenType fonts. |
| otfRomanItalics | <code>boolean</code> otfRomanItalics | If true, applies italics to half-width alphanumerics. |
| paragraphGyoudori | <code>boolean</code> paragraphGyoudori | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. |
| preferences | <code>{Preferences}</code> *preferences | A collection of preferences objects. |
| rensuuji | <code>boolean</code> rensuuji | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. |
| rotateSingleByteCharacters | <code>boolean</code> rotateSingleByteCharacters | If true, rotates Roman characters in vertical text. |

| CHARACTER CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|--|---|
| <code>rubyAlignment</code> | <code>#RubyAlignments rubyAlignment</code> | <i>The ruby alignment.</i> |
| <code>rubyAutoAlign</code> | <code>boolean rubyAutoAlign</code> | <i>If true, auto aligns ruby.</i> |
| <code>rubyAutoScaling</code> | <code>boolean rubyAutoScaling</code> | <i>If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent.</i> |
| <code>rubyAutoTcyAutoScale</code> | <code>boolean rubyAutoTcyAutoScale</code> | <i>If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em.</i> |
| <code>rubyAutoTcyDigits</code> | <code>const rubyAutoTcyDigits</code> | <i>The number of digits included in auto tcy (tate-chuu-yoko) in ruby.</i> |
| <code>rubyAutoTcyIncludeRoman</code> | <code>boolean rubyAutoTcyIncludeRoman</code> | <i>If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby.</i> |
| <code>rubyFill</code> | <code>variant rubyFill</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch or String.</i> |
| <code>rubyFlag</code> | <code>boolean rubyFlag</code> | <i>If true, ruby is on.</i> |
| <code>rubyFont</code> | <code>variant rubyFont</code> | <i>The font applied to ruby characters. Type: Font or String.</i> |
| <code>rubyFontSize</code> | <code>double rubyFontSize</code> | <i>The size (in points) of ruby characters.</i> |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | <i>The font style of ruby characters. Type: String or idNothingEnum enumerator.</i> |
| <code>rubyOpenTypePro</code> | <code>boolean rubyOpenTypePro</code> | <i>If true, uses OpenType Pro fonts for ruby.</i> |
| <code>rubyOverhang</code> | <code>boolean rubyOverhang</code> | <i>If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount.</i> |
| <code>rubyOverprintFill</code> | <code>#AdornmentOverprint rubyOverprintFill</code> | <i>The method of overprinting the ruby fill.</i> |
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | <i>The method of overprinting the ruby stroke.</i> |
| <code>rubyParentOverhangAmount</code> | <code>#RubyOverhang rubyParentOverhangAmount</code> | <i>The amount by which ruby characters can overhang the parent text.</i> |
| <code>rubyParentScalingPercent</code> | <code>double rubyParentScalingPercent</code> | <i>The amount (as a percentage) to scale the parent text size to determine the ruby text size.</i> |
| <code>rubyParentSpacing</code> | <code>#RubyParentSpacing rubyParentSpacing</code> | <i>The ruby spacing relative to the parent text.</i> |
| <code>rubyPosition</code> | <code>#RubyKetenPosition rubyPosition</code> | <i>The position of ruby characters relative to the parent text.</i> |
| <code>rubyString</code> | <code>string rubyString</code> | <i>The ruby string contents.</i> |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String.</i> |
| <code>rubyStrokeTint</code> | <code>double rubyStrokeTint</code> | <i>The stroke tint (as a percentage) of ruby characters.</i> |
| <code>rubyTint</code> | <code>double rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100).</i> |
| <code>rubyType</code> | <code>#RubyTypes rubyType</code> | <i>The ruby type.</i> |
| <code>rubyWeight</code> | <code>double rubyWeight</code> | <i>The stroke weight (in points) of ruby characters.</i> |
| <code>rubyXOffset</code> | <code>double rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters.</i> |
| <code>rubyXScale</code> | <code>double rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size.</i> |
| <code>rubyYOffset</code> | <code>double rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters.</i> |
| <code>rubyYScale</code> | <code>double rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size.</i> |
| <code>scaleAffectsLineHeight</code> | <code>boolean scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled.</i> |

| CHARACTER CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>shataiAdjustRotation</code> | <code>boolean shataiAdjustRotation</code> | If true, applies shatai rotation. |
| <code>shataiAdjustTsume</code> | <code>boolean shataiAdjustTsume</code> | If true, adjusts shatai tsume. |
| <code>shataiDegreeAngle</code> | <code>double shataiDegreeAngle</code> | The shatai lens angle (in degrees). |
| <code>shataiMagnification</code> | <code>double shataiMagnification</code> | The amount (as a percentage) of shatai obliquing to apply. |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | The minimum space after a span or a split column. Type: Unit (Double or String). |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | The minimum space before a span or a split column. Type: Unit (Double or String). |
| <code>spanColumnType</code> | <code>#SpanColumnTypeOptions spanColumnType</code> | Whether a paragraph should be a single column, span columns or split columns. |
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator. |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | The inside gutter if the paragraph splits columns. Type: Unit (Double or String). |
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | The outside gutter if the paragraph splits columns. Type: Unit (Double or String). |
| <code>tatechuyoko</code> | <code>boolean tatechuyoko</code> | If true, makes the character horizontal in vertical text. |
| <code>tatechuyokoXOffset</code> | <code>double tatechuyokoXOffset</code> | The horizontal offset for horizontal characters in vertical text. |
| <code>tatechuyokoYOffset</code> | <code>double tatechuyokoYOffset</code> | The vertical offset for horizontal characters in vertical text. |
| <code>trailingAki</code> | <code>double trailingAki</code> | The amount of space after each character. |
| <code>treatIdeographicSpaceAsSpace</code> | <code>boolean treatIdeographicSpaceAsSpace</code> | If true, ideographic spaces will not wrap to the next line like text characters. |
| <code>tsume</code> | <code>double tsume</code> | The amount of horizontal character compression. |
| <code>warichu</code> | <code>boolean warichu</code> | If true, turns on warichu. |
| <code>warichuAlignment</code> | <code>#WarichuAlignment warichuAlignment</code> | The warichu alignment. |
| <code>warichuCharsAfterBreak</code> | <code>const warichuCharsAfterBreak</code> | The minimum number of characters allowed after a line break. |
| <code>warichuCharsBeforeBreak</code> | <code>const warichuCharsBeforeBreak</code> | The minimum number of characters allowed before a line break. |
| <code>warichuLines</code> | <code>const warichuLines</code> | The number of lines of warichu within a single normal line. |
| <code>warichuLineSpacing</code> | <code>double warichuLineSpacing</code> | The gap between lines of warichu characters. |
| <code>warichuSize</code> | <code>double warichuSize</code> | The amount (as a percentage) to scale parent text size to determine warichu size. |

CharacterStyle

Object available in ID 3, 4, 5, 6, 7

| CHARACTERSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---|--|
| <code>characterAlignment</code> | <code>variant characterAlignment</code> | The alignment of small characters to the largest character in the line. Type: idCharacterAlignment enumerator or idNothingEnum enumerator. |

| CHARACTERSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|---|
| characterRotation | <code>variant characterRotation</code> | <i>The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. Type: Double or idNothingEnum enumerator.</i> |
| cjkGridTracking | <code>variant cjkGridTracking</code> | <i>If true, uses grid tracking to track non-Roman characters in CJK grids. Type: Boolean or idNothingEnum enumerator.</i> |
| glyphForm | <code>variant glyphForm</code> | <i>The glyph variant to substitute for standard glyphs. Type: idAlternateGlyphForms enumerator or idNothingEnum enumerator.</i> |
| jidori | <code>variant jidori</code> | <i>The number of grid squares in which to arrange the text. . Type: Integer or idNothingEnum enumerator.</i> |
| kentenAlignment | <code>variant kentenAlignment</code> | <i>The alignment of kenten characters relative to the parent characters. . Type: idKntenAlignment enumerator or idNothingEnum enumerator.</i> |
| kentenCharacterSet | <code>variant kentenCharacterSet</code> | <i>The character set used for the custom kenten character. Note: Valid only when kenten kind is custom. . Type: idKntenCharacterSet enumerator or idNothingEnum enumerator.</i> |
| kentenCustomCharacter | <code>variant kentenCustomCharacter</code> | <i>The character used for kenten. Note: Valid only when kenten kind is custom. Type: String or idNothingEnum enumerator.</i> |
| kentenFillColor | <code>variant kentenFillColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch, String or idNothingEnum enumerator.</i> |
| kentenFont | <code>variant kentenFont</code> | <i>The font to use for kenten characters. Type: Font, String or idNothingEnum enumerator.</i> |
| kentenFontSize | <code>variant kentenFontSize</code> | <i>The size (in points) of kenten characters. Type: Double or idNothingEnum enumerator.</i> |
| kentenFontStyle | <code>variant kentenFontStyle</code> | <i>The font style of kenten characters. Type: String or idNothingEnum enumerator.</i> |
| kentenKind | <code>variant kentenKind</code> | <i>The style of kenten characters. Type: idKntenCharacter enumerator or idNothingEnum enumerator.</i> |
| kentenOverprintFill | <code>variant kentenOverprintFill</code> | <i>The method of overprinting the kenten fill. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator.</i> |
| kentenOverprintStroke | <code>variant kentenOverprintStroke</code> | <i>The method of overprinting the kenten stroke. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator.</i> |
| kentenPlacement | <code>variant kentenPlacement</code> | <i>The distance between kenten characters and their parent characters. Type: Double or idNothingEnum enumerator.</i> |
| kentenPosition | <code>variant kentenPosition</code> | <i>The kenten position relative to the parent character. Type: idRubyKtentenPosition enumerator or idNothingEnum enumerator.</i> |
| kentenStrokeColor | <code>variant kentenStrokeColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of kenten characters. Type: Swatch, String or idNothingEnum enumerator.</i> |
| kentenStrokeTint | <code>variant kentenStrokeTint</code> | <i>The stroke tint (as a percentage) of kenten characters. (Range: 0 to 100). Type: Double or idNothingEnum enumerator.</i> |
| kentenTint | <code>variant kentenTint</code> | <i>The fill tint (as a percentage) of kenten characters. (Range: 0 to 100). Type: Double or idNothingEnum enumerator.</i> |
| kentenWeight | <code>variant kentenWeight</code> | <i>The stroke weight (in points) of kenten characters. Type: Double or idNothingEnum enumerator.</i> |
| kentenXScale | <code>variant kentenXScale</code> | <i>The horizontal size of kenten characters as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| kentenYScale | <code>variant kentenYScale</code> | <i>The vertical size of kenten characters as a percent of the original size. . Type: Double or idNothingEnum enumerator.</i> |

| CHARACTERSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---|--|
| leadingAki | <code>variant leadingAki</code> | <i>The amount of space before each character. Type: Double or idNothingEnum enumerator.</i> |
| otfHVKana | <code>variant otfHVKana</code> | <i>If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. . Type: Boolean or idNothingEnum enumerator.</i> |
| otfProportionalMetrics | <code>variant otfProportionalMetrics</code> | <i>If true, kerns according to proportional CJK metrics in OpenType fonts. Type: Boolean or idNothingEnum enumerator.</i> |
| otfRomanItalics | <code>variant otfRomanItalics</code> | <i>If true, applies italics to half-width alphanumerics. . Type: Boolean or idNothingEnum enumerator.</i> |
| rubyAlignment | <code>variant rubyAlignment</code> | <i>The ruby alignment. Type: idRubyAlignments enumerator or idNothingEnum enumerator.</i> |
| rubyAutoAlign | <code>variant rubyAutoAlign</code> | <i>If true, auto aligns ruby. Type: Boolean or idNothingEnum enumerator.</i> |
| rubyAutoScaling | <code>variant rubyAutoScaling</code> | <i>If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. . Type: Boolean or idNothingEnum enumerator.</i> |
| rubyAutoTcyAutoScale | <code>variant rubyAutoTcyAutoScale</code> | <i>If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. Type: Boolean or idNothingEnum enumerator.</i> |
| rubyAutoTcyDigits | <code>variant rubyAutoTcyDigits</code> | <i>The number of digits included in auto tcy (tate-chuu-yoko) in ruby. Type: Integer or idNothingEnum enumerator.</i> |
| rubyAutoTcyIncludeRoman | <code>variant rubyAutoTcyIncludeRoman</code> | <i>If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. Type: Boolean or idNothingEnum enumerator.</i> |
| rubyFill | <code>variant rubyFill</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch, String or idNothingEnum enumerator.</i> |
| rubyFont | <code>variant rubyFont</code> | <i>The font applied to ruby characters. Type: Font, String or idNothingEnum enumerator.</i> |
| rubyFontSize | <code>variant rubyFontSize</code> | <i>The size (in points) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyFontStyle | <code>variant rubyFontStyle</code> | <i>The font style of ruby characters. Type: String or idNothingEnum enumerator.</i> |
| rubyOpenTypePro | <code>variant rubyOpenTypePro</code> | <i>If true, uses OpenType Pro fonts for ruby. Type: Boolean or idNothingEnum enumerator.</i> |
| rubyOverhang | <code>variant rubyOverhang</code> | <i>If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. . Type: Boolean or idNothingEnum enumerator.</i> |
| rubyOverprintFill | <code>variant rubyOverprintFill</code> | <i>The method of overprinting the ruby fill. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator.</i> |
| rubyOverprintStroke | <code>variant rubyOverprintStroke</code> | <i>The method of overprinting the ruby stroke. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator.</i> |
| rubyParentOverhangAmount | <code>variant rubyParentOverhangAmount</code> | <i>The amount by which ruby characters can overhang the parent text. Type: idRubyOverhang enumerator or idNothingEnum enumerator.</i> |
| rubyParentScalingPercent | <code>variant rubyParentScalingPercent</code> | <i>The amount (as a percentage) to scale the parent text size to determine the ruby text size. Type: Double or idNothingEnum enumerator.</i> |
| rubyParentSpacing | <code>variant rubyParentSpacing</code> | <i>The ruby spacing relative to the parent text. . Type: idRubyParentSpacing enumerator or idNothingEnum enumerator.</i> |
| rubyPosition | <code>variant rubyPosition</code> | <i>The position of ruby characters relative to the parent text. Type: idRubyKetenPosition enumerator or idNothingEnum enumerator.</i> |

| CHARACTERSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|---|
| rubyStroke | <code>variant rubyStroke</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch, String or idNothingEnum enumerator.</i> |
| rubyStrokeTint | <code>variant rubyStrokeTint</code> | <i>The stroke tint (as a percentage) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyTint | <code>variant rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). Type: Double or idNothingEnum enumerator.</i> |
| rubyType | <code>variant rubyType</code> | <i>The ruby type. Type: idRubyTypes enumerator or idNothingEnum enumerator.</i> |
| rubyWeight | <code>variant rubyWeight</code> | <i>The stroke weight (in points) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyXOffset | <code>variant rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyXScale | <code>variant rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| rubyYOffset | <code>variant rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyYScale | <code>variant rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| scaleAffectsLineHeight | <code>variant scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled. . Type: Boolean or idNothingEnum enumerator.</i> |
| shataiAdjustRotation | <code>variant shataiAdjustRotation</code> | <i>If true, applies shatai rotation. Type: Boolean or idNothingEnum enumerator.</i> |
| shataiAdjustTsume | <code>variant shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume. Type: Boolean or idNothingEnum enumerator.</i> |
| shataiDegreeAngle | <code>variant shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees). Type: Double or idNothingEnum enumerator.</i> |
| shataiMagnification | <code>variant shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply. Type: Double or idNothingEnum enumerator.</i> |
| tatechuyoko | <code>variant tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text. Type: Boolean or idNothingEnum enumerator.</i> |
| tatechuyokoXOffset | <code>variant tatechuyokoXOffset</code> | <i>The horizontal offset for horizontal characters in vertical text. Type: Double or idNothingEnum enumerator.</i> |
| tatechuyokoYOffset | <code>variant tatechuyokoYOffset</code> | <i>The vertical offset for horizontal characters in vertical text. Type: Double or idNothingEnum enumerator.</i> |
| trailingAki | <code>variant trailingAki</code> | <i>The amount of space after each character. Type: Double or idNothingEnum enumerator.</i> |
| tsume | <code>variant tsume</code> | <i>The amount of horizontal character compression. Type: Double or idNothingEnum enumerator.</i> |
| warichu | <code>variant warichu</code> | <i>If true, turns on warichu. Type: Boolean or idNothingEnum enumerator.</i> |
| warichuAlignment | <code>variant warichuAlignment</code> | <i>The warichu alignment. Type: idWarichuAlignment enumerator or idNothingEnum enumerator.</i> |
| warichuCharsAfterBreak | <code>variant warichuCharsAfterBreak</code> | <i>The minimum number of characters allowed after a line break. . Type: Integer or idNothingEnum enumerator.</i> |
| warichuCharsBeforeBreak | <code>variant warichuCharsBeforeBreak</code> | <i>The minimum number of characters allowed before a line break. Type: Integer or idNothingEnum enumerator.</i> |

| CHARACTERSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---|--|
| <code>warichuLines</code> | <code>variant warichuLines</code> | <i>The number of lines of warichu within a single normal line. Type: Integer or idNothingEnum enumerator.</i> |
| <code>warichuLineSpacing</code> | <code>variant warichuLineSpacing</code> | <i>The gap between lines of warichu characters. Type: Double or idNothingEnum enumerator.</i> |
| <code>warichuSize</code> | <code>variant warichuSize</code> | <i>The amount (as a percentage) to scale parent text size to determine warichu size. Type: Double or idNothingEnum enumerator.</i> |

Column

Object available in ID 3, 4, 5, 6, 7

| COLUMN CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|---|
| <code>changeTransliterate</code> | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | <i>Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean.</i> |
| <code>findTransliterate</code> | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | <i>Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean.</i> |
| <code>writingDirection</code> | <code>#HorizontalOrVertical writingDirection</code> | <i>The direction of the text in the cell.</i> |

CrossReference

Object available in ID 3, 4, 5, 6, 7

| CROSSREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--------------------------|---|
| <code>name</code> | <code>string name</code> | <i>The name of the CrossReference; this is an alias to the CrossReference's label property.</i> |

Document

Object available in ID 3, 4, 5, 6, 7

| DOCUMENT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| <code>changeComposer</code> | <code>void changeComposer()</code> | <i>Change comoser to optyca.</i> |

| DOCUMENT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|--|
| <code>changeTransliterate</code> | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| <code>clearFrameFittingOptions</code> | <code>void clearFrameFittingOptions()</code> | Removes the frame fittings options and resets it to the initial state. |
| <code>findTransliterate</code> | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| <code>resetAllButtons</code> | <code>void resetAllButtons()</code> | Resets all the buttons to their Normal state. |
| <code>resetAllMultiStateObjects</code> | <code>void resetAllMultiStateObjects()</code> | Resets all the multi-state objects to their first state. |
| <code>saveACopy</code> | <code>void saveACopy(variant To, boolean Stationery)</code> | Saves a copy of the document. / To: The file path for the copy. Note: Leaves the original document open and does not open the copy. Type: File (String). / Stationery: If true, saves the file as stationery (Mac OS) or as a template (Windows). Note: The file extension for stationery and templates is different than the extension for regular files. |
| <code>cjkGridPreferences</code> | <code>{CjkGridPreference} *cjkGridPreferences</code> | CJK grid preference settings. |
| <code>compositeFonts</code> | <code>{CompositeFonts} *compositeFonts</code> | A collection of composite fonts. |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| <code>id</code> | <code>integer *id</code> | The unique ID of the Document. |
| <code>kinsokuTables</code> | <code>{KinsokuTables} *kinsokuTables</code> | A collection of kinsoku tables. |
| <code>layoutGridData</code> | <code>{LayoutGridDataInformation} *layoutGridData</code> | Default properties specific to layout grids. |
| <code>mojikumiTables</code> | <code>{MojikumiTables} *mojikumiTables</code> | A collection of mojikumi tables. |
| <code>mojikumiUIPreferences</code> | <code>{MojikumiUiPreference} *mojikumiUIPreferences</code> | Mojikumi user interface preference settings. |
| <code>motionPresets</code> | <code>{MotionPresets} *motionPresets</code> | A collection of Motion presets. |
| <code>multiStateObjects</code> | <code>{MultiStateObjects} *multiStateObjects</code> | A collection of multi-state objects. |
| <code>namedGrids</code> | <code>{NamedGrids} *namedGrids</code> | A collection of named grids. |
| <code>placeGuns</code> | <code>{PlaceGun} *placeGuns</code> | The place gun. |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| <code>storyGridData</code> | <code>{StoryGridDataInformation} *storyGridData</code> | Default properties specific to frame grids. |
| <code>watermarkPreferences</code> | <code>{WatermarkPreference} *watermarkPreferences</code> | Watermark preferences. |

DocumentPreference

Object available in ID 3, 4, 5, 6, 7

| DOCUMENTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|---|
| <code>intent</code> | <code>#DocumentIntentOptions intent</code> | The intent for a document. |
| <code>startPageNumber</code> | <code>integer startPageNumber</code> | The starting page number for a document. This is the same as the starting page number for the first section of a document. Default value is 1. Type: Long Integer (1 - 999999). |

DocumentPreset

Object available in ID 3, 4, 5, 6, 7

| DOCUMENTPRESET CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|--|
| <code>intent</code> | <code>#DocumentIntentOptions intent</code> | <i>The intent for a document.</i> |
| <code>startPageNumber</code> | <code>integer startPageNumber</code> | <i>The starting page number for a document. This is the same as the starting page number for the first section of a document. Default value is 1. Type: Long Integer (1 - 999999).</i> |

Documents

Object available in ID 3, 4, 5, 6, 7

| DOCUMENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------|---|--|
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | <i>Returns the object with the specified ID. / Return value: The object. / ID: The ID.</i> |

EPS

Object available in ID 3, 4, 5, 6, 7

| EPS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| <code>clearTransformations</code> | <code>void clearTransformations()</code> | <i>Clears transformations from the EPS. Transformations include rotation, scaling, flipping, fitting, and shearing.</i> |
| <code>convertShape</code> | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | <i>Converts the EPS to a different shape. / Given: The EPS's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0). Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points).</i> |
| <code>select</code> | <code>void select(#SelectionOptions ExistingSelection)</code> | <i>Selects the object. / ExistingSelection: The selection status of the EPS in relation to previously selected objects.</i> |
| <code>allGraphics</code> | <code>{Objects} *allGraphics</code> | <i>Lists all graphics contained by the EPS.</i> |
| <code>allPageItems</code> | <code>{Objects} *allPageItems</code> | <i>Lists all page items contained by the EPS.</i> |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | <i>The page item animation settings.</i> |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |

| EPS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|---|--|
| bottomLeftCornerRadius | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| bottomRightCornerOption | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| bottomRightCornerRadius | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| endCap | <code>#EndCap endCap</code> | The end shape of an open path. |
| endJoin | <code>#EndJoin endJoin</code> | The corner join applied to the EPS. |
| fillColor | <code>variant fillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of the EPS.. Type: Swatch or String. |
| fillTint | <code>double fillTint</code> | The percent of tint to use in the EPS's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| gapColor | <code>{Swatch} gapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type. |
| gapTint | <code>double gapTint</code> | The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| gradientFillAngle | <code>double gradientFillAngle</code> | The angle of a linear gradient applied to the fill of the EPS. (Range: -180 to 180). |
| gradientFillLength | <code>variant gradientFillLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the EPS. Type: Unit (Double or String). |
| gradientFillStart | <code>variant gradientFillStart</code> | The starting point (in page coordinates) of a gradient applied to the fill of the EPS, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| gradientStrokeAngle | <code>double gradientStrokeAngle</code> | The angle of a linear gradient applied to the stroke of the EPS. (Range: -180 to 180). |
| gradientStrokeLength | <code>variant gradientStrokeLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the EPS. Type: Unit (Double or String). |
| gradientStrokeStart | <code>variant gradientStrokeStart</code> | The starting point (in page coordinates) of a gradient applied to the stroke of the EPS, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| itemLayer | <code>{Layer} itemLayer</code> | The layer that the EPS is on. |
| leftLineEnd | <code>#ArrowHead leftLineEnd</code> | The arrowhead applied to the start of the path. |
| locked | <code>boolean locked</code> | If true, the EPS is locked. |
| miterLimit | <code>double miterLimit</code> | The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500). |
| name | <code>string name</code> | The name of the EPS. |
| overprintFill | <code>boolean overprintFill</code> | If true, the EPS's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors. |
| overprintGap | <code>boolean overprintGap</code> | If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors. |
| overprintStroke | <code>boolean overprintStroke</code> | If true, the EPS's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors. |
| parentPage | <code>{Page} *parentPage</code> | The page on which this page item appears. |
| preferences | <code>{Preferences} *preferences</code> | A collection of preferences objects. |

| EPS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|---|--|
| <code>rightLineEnd</code> | <code>#ArrowHead rightLineEnd</code> | <i>The arrowhead applied to the end of the path.</i> |
| <code>strokeAlignment</code> | <code>#StrokeAlignment strokeAlignment</code> | <i>The stroke alignment applied to the EPS.</i> |
| <code>strokeColor</code> | <code>variant strokeColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the EPS. Type: Swatch or String.</i> |
| <code>strokeCornerAdjustment</code> | <code>#StrokeCornerAdjustment strokeCornerAdjustment</code> | <i>The corner adjustment applied to the EPS.</i> |
| <code>strokeDashAndGap</code> | <code>variant strokeDashAndGap</code> | <i>The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings).</i> |
| <code>strokeTint</code> | <code>double strokeTint</code> | <i>The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>strokeType</code> | <code>variant strokeType</code> | <i>The name of the stroke style to apply. Type: StrokeStyle or String.</i> |
| <code>strokeWeight</code> | <code>variant strokeWeight</code> | <i>The weight (in points) to apply to the EPS's stroke. Type: Unit (Double or String).</i> |
| <code>timingSettings</code> | <code>{TimingSetting} •timingSettings</code> | <i>The object timing settings.</i> |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the EPS is visible.</i> |

Font

Object available in ID 3, 4, 5, 6, 7

| FONT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|--|--|
| <code>createSubsetFont</code> | <code>void createSubsetFont(string CharactersForSubset, string FontDestination)</code> | <i>Create a copy of the font with just enough information to render the list of characters given. / CharactersForSubset: String with all the characters used in the resulting font. / FontDestination: File location for the new font.</i> |

FormField

Object available in ID 3, 4, 5, 6, 7

| FORMFIELD CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|--|
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| activeStateIndex | <code>integer activeStateIndex</code> | The index of the active state in the object's states collection. |
| animationSettings | <code>{AnimationSetting} *animationSettings</code> | The page item animation settings. |
| bottomLeftCornerOption | <code>#CornerOptions bottomLeftCornerOption</code> | The shape to apply to the bottom left corner of rectangular shapes. |
| bottomLeftCornerRadius | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| bottomRightCornerOption | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| bottomRightCornerRadius | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| parentPage | <code>{Page} *parentPage</code> | The page on which this page item appears. |
| preferences | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| timingSettings | <code>{TimingSetting} *timingSettings</code> | The object timing settings. |
| topLeftCornerOption | <code>#CornerOptions topLeftCornerOption</code> | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. |
| topLeftCornerRadius | <code>variant topLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String). |
| topRightCornerOption | <code>#CornerOptions topRightCornerOption</code> | The shape to apply to the top right corner of rectangular shapes. |
| topRightCornerRadius | <code>variant topRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String). |
| visible | <code>boolean visible</code> | If true, the FormField is visible. |

FormFields

Object available in ID 3, 4, 5, 6, 7

| FORMFIELDS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------|---|--|
| <code>add</code> | <code>{FormField} add(variant Layer, #LocationOptions At, variant Reference)</code> | <i>Creates a new FormField. / Return value: The new FormField. / Layer: The layer on which to create the FormField. Type: Layer. / At: The location at which to insert the FormField relative to the reference object or within the container object. / Reference: The reference object. Note: Required when the at parameter is before or after. Type: Document, Spread, MasterSpread, Page, Layer or Pageltem.</i> |

GeneralPreference

Object available in ID 3, 4, 5, 6, 7

| GENERALPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|---|
| <code>enableMultiTouchGestures</code> | <code>boolean enableMultiTouchGestures</code> | <i>Controls whether or not multi-touch gestures are enabled.</i> |
| <code>objectsMoveWithPage</code> | <code>boolean objectsMoveWithPage</code> | <i>Controls whether page items move when a page is repositioned from the UI. The option/alt key temporarily reverses this property.</i> |
| <code>preventSelectingLockedItems</code> | <code>boolean preventSelectingLockedItems</code> | <i>Controls whether or not you can select and interact with a locked item. When this is off, only position is locked.</i> |
| <code>showContentGrabber</code> | <code>boolean showContentGrabber</code> | <i>Controls whether or not the content grabber adornment is shown.</i> |
| <code>showLiveCorners</code> | <code>boolean showLiveCorners</code> | <i>Controls whether or not the live corners grabber adornment is shown.</i> |
| <code>showMasterPageOverlay</code> | <code>boolean showMasterPageOverlay</code> | <i>Controls whether or not to show the master page overlay when a page is selected using the Page Tool.</i> |

Graphic

Object available in ID 3, 4, 5, 6, 7

| GRAPHIC CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| <code>clearTransformations</code> | <code>void clearTransformations()</code> | <i>Clears transformations from the Graphic. Transformations include rotation, scaling, flipping, fitting, and shearing.</i> |

| GRAPHIC CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|---|
| convertShape | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | Converts the Graphic to a different shape. / Given: The Graphic's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0) . Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points). |
| select | <code>void select(#SelectionOptions ExistingSelection)</code> | Selects the object. / ExistingSelection: The selection status of the Graphic in relation to previously selected objects. |
| store | <code>{Asset} store(Library Using)</code> | Stores the object in the specified library. / Return value: The new library asset(s). / Using: The library in which to store the object. |
| allGraphics | <code>{Objects} *allGraphics</code> | Lists all graphics contained by the Graphic. |
| allPageItems | <code>{Objects} *allPageItems</code> | Lists all page items contained by the Graphic. |
| animationSettings | <code>{AnimationSetting} *animationSettings</code> | The page item animation settings. |
| bottomLeftCornerOption | <code>#CornerOptions bottomLeftCornerOption</code> | The shape to apply to the bottom left corner of rectangular shapes. |
| bottomLeftCornerRadius | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| bottomRightCornerOption | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| bottomRightCornerRadius | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| contentTransparencySettings | <code>{ContentTransparencySetting} *contentTransparencySettings</code> | Transparency settings for the content of the Graphic. |
| endCap | <code>#EndCap endCap</code> | The end shape of an open path. |
| endJoin | <code>#EndJoin endJoin</code> | The corner join applied to the Graphic. |
| fillColor | <code>variant fillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of the Graphic. . Type: Swatch or String. |
| fillTint | <code>double fillTint</code> | The percent of tint to use in the Graphic's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| fillTransparencySettings | <code>{FillTransparencySetting} *fillTransparencySettings</code> | Transparency settings for the fill applied to the Graphic. |
| gapColor | <code>{Swatch} gapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type. |
| gapTint | <code>double gapTint</code> | The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| gradientFillAngle | <code>double gradientFillAngle</code> | The angle of a linear gradient applied to the fill of the Graphic. (Range: -180 to 180). |
| gradientFillLength | <code>variant gradientFillLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the Graphic. Type: Unit (Double or String). |
| gradientFillStart | <code>variant gradientFillStart</code> | The starting point (in page coordinates) of a gradient applied to the fill of the Graphic, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| gradientStrokeAngle | <code>double gradientStrokeAngle</code> | The angle of a linear gradient applied to the stroke of the Graphic. (Range: -180 to 180). |
| gradientStrokeLength | <code>variant gradientStrokeLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the Graphic. Type: Unit (Double or String). |

| GRAPHIC CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>gradientStrokeStart</code> | <code>variant gradientStrokeStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the stroke of the Graphic, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>itemLayer</code> | <code>{Layer} itemLayer</code> | <i>The layer that the Graphic is on.</i> |
| <code>leftLineEnd</code> | <code>#ArrowHead leftLineEnd</code> | <i>The arrowhead applied to the start of the path.</i> |
| <code>locked</code> | <code>boolean locked</code> | <i>If true, the Graphic is locked.</i> |
| <code>miterLimit</code> | <code>double miterLimit</code> | <i>The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500).</i> |
| <code>name</code> | <code>string name</code> | <i>The name of the Graphic.</i> |
| <code>nonprinting</code> | <code>boolean nonprinting</code> | <i>If true, the Graphic does not print.</i> |
| <code>overprintFill</code> | <code>boolean overprintFill</code> | <i>If true, the Graphic's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors.</i> |
| <code>overprintGap</code> | <code>boolean overprintGap</code> | <i>If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors.</i> |
| <code>overprintStroke</code> | <code>boolean overprintStroke</code> | <i>If true, the Graphic's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors.</i> |
| <code>parentPage</code> | <code>{Page} *parentPage</code> | <i>The page on which this page item appears.</i> |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| <code>rightLineEnd</code> | <code>#ArrowHead rightLineEnd</code> | <i>The arrowhead applied to the end of the path.</i> |
| <code>strokeAlignment</code> | <code>#StrokeAlignment strokeAlignment</code> | <i>The stroke alignment applied to the Graphic.</i> |
| <code>strokeColor</code> | <code>variant strokeColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the Graphic. Type: Swatch or String.</i> |
| <code>strokeCornerAdjustment</code> | <code>#StrokeCornerAdjustment strokeCornerAdjustment</code> | <i>The corner adjustment applied to the Graphic.</i> |
| <code>strokeDashAndGap</code> | <code>variant strokeDashAndGap</code> | <i>The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings).</i> |
| <code>strokeTint</code> | <code>double strokeTint</code> | <i>The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>strokeTransparencySettings</code> | <code>{StrokeTransparencySetting} *strokeTransparencySettings</code> | <i>Transparency settings for the stroke.</i> |
| <code>strokeType</code> | <code>variant strokeType</code> | <i>The name of the stroke style to apply. Type: StrokeStyle or String.</i> |
| <code>strokeWeight</code> | <code>variant strokeWeight</code> | <i>The weight (in points) to apply to the Graphic's stroke. Type: Unit (Double or String).</i> |
| <code>textWrapPreferences</code> | <code>{TextWrapPreference} *textWrapPreferences</code> | <i>The text wrap preference properties that define the default formatting for wrapping text around objects.</i> |
| <code>timingSettings</code> | <code>{TimingSetting} *timingSettings</code> | <i>The object timing settings.</i> |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |

| GRAPHIC CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|---|
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>transparencySettings</code> | <code>{TransparencySetting} *transparencySettings</code> | <i>Transparency settings.</i> |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the Graphic is visible.</i> |

GraphicLine

Object available in ID 3, 4, 5, 6, 7

| GRAPHICLINE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | <i>The page item animation settings.</i> |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>buttons</code> | <code>{Buttons} *buttons</code> | <i>A collection of buttons.</i> |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |
| <code>formFields</code> | <code>{FormFields} *formFields</code> | <i>A collection of form fields.</i> |
| <code>mediaItems</code> | <code>{MediaItems} *mediaItems</code> | <i>The media items collection.</i> |
| <code>movies</code> | <code>{Movies} *movies</code> | <i>A collection of movies.</i> |
| <code>multiStateObjects</code> | <code>{MultiStateObjects} *multiStateObjects</code> | <i>A collection of multi-state objects.</i> |
| <code>name</code> | <code>string name</code> | <i>The name of the GraphicLine.</i> |
| <code>parentPage</code> | <code>{Page} *parentPage</code> | <i>The page on which this page item appears.</i> |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| <code>sounds</code> | <code>{Sounds} *sounds</code> | <i>A collection of sound clips.</i> |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |
| <code>timingSettings</code> | <code>{TimingSetting} *timingSettings</code> | <i>The object timing settings.</i> |

| GRAPHICLINE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|--|--|
| topLeftCornerOption | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| topLeftCornerRadius | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| topRightCornerOption | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| topRightCornerRadius | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| visible | <code>boolean visible</code> | <i>If true, the GraphicLine is visible.</i> |

Group

Object available in ID 3, 4, 5, 6, 7

| GROUP CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|---|
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| animationSettings | <code>{AnimationSetting} *animationSettings</code> | <i>The page item animation settings.</i> |
| bottomLeftCornerOption | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |
| bottomLeftCornerRadius | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String).</i> |
| bottomRightCornerOption | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| bottomRightCornerRadius | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| epsTexts | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |
| multiStateObjects | <code>{MultiStateObjects} *multiStateObjects</code> | <i>A collection of multi-state objects.</i> |
| name | <code>string name</code> | <i>The name of the Group.</i> |
| parentPage | <code>{Page} *parentPage</code> | <i>The page on which this page item appears.</i> |
| preferences | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| splineItems | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |
| timingSettings | <code>{TimingSetting} *timingSettings</code> | <i>The object timing settings.</i> |
| topLeftCornerOption | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |

| GROUP CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|---|
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes.</i> Type: Unit (Double or String). |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes.</i> Type: Unit (Double or String). |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the Group is visible.</i> |

Guide

Object available in ID 3, 4, 5, 6, 7

| GUIDE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------|---------------------------------|---|
| <code>name</code> | <code>string name</code> | <i>The name of the Guide; this is an alias to the Guide's label property.</i> |
| <code>parentPage</code> | <code>{Page} *parentPage</code> | <i>The page on which this page item appears.</i> |

Image

Object available in ID 3, 4, 5, 6, 7

| IMAGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file.</i> / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| <code>clearTransformations</code> | <code>void clearTransformations()</code> | <i>Clears transformations from the Image. Transformations include rotation, scaling, flipping, fitting, and shearing.</i> |
| <code>convertShape</code> | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | <i>Converts the Image to a different shape.</i> / Given: The Image's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0). Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points). |
| <code>select</code> | <code>void select(#SelectionOptions ExistingSelection)</code> | <i>Selects the object.</i> / ExistingSelection: The selection status of the Image in relation to previously selected objects. |
| <code>allGraphics</code> | <code>{Objects} *allGraphics</code> | <i>Lists all graphics contained by the Image.</i> |
| <code>allPageItems</code> | <code>{Objects} *allPageItems</code> | <i>Lists all page items contained by the Image.</i> |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | <i>The page item animation settings.</i> |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |

| IMAGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|---|--|
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>endCap</code> | <code>#EndCap endCap</code> | <i>The end shape of an open path.</i> |
| <code>endJoin</code> | <code>#EndJoin endJoin</code> | <i>The corner join applied to the Image.</i> |
| <code>gapColor</code> | <code>{Swatch} gapColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type.</i> |
| <code>gapTint</code> | <code>double gapTint</code> | <i>The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>gradientFillAngle</code> | <code>double gradientFillAngle</code> | <i>The angle of a linear gradient applied to the fill of the Image. (Range: -180 to 180).</i> |
| <code>gradientFillLength</code> | <code>variant gradientFillLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the Image. Type: Unit (Double or String).</i> |
| <code>gradientFillStart</code> | <code>variant gradientFillStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the fill of the Image, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>gradientStrokeAngle</code> | <code>double gradientStrokeAngle</code> | <i>The angle of a linear gradient applied to the stroke of the Image. (Range: -180 to 180).</i> |
| <code>gradientStrokeLength</code> | <code>variant gradientStrokeLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the Image. Type: Unit (Double or String).</i> |
| <code>gradientStrokeStart</code> | <code>variant gradientStrokeStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the stroke of the Image, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>itemLayer</code> | <code>{Layer} itemLayer</code> | <i>The layer that the Image is on.</i> |
| <code>leftLineEnd</code> | <code>#ArrowHead leftLineEnd</code> | <i>The arrowhead applied to the start of the path.</i> |
| <code>locked</code> | <code>boolean locked</code> | <i>If true, the Image is locked.</i> |
| <code>miterLimit</code> | <code>double miterLimit</code> | <i>The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500).</i> |
| <code>name</code> | <code>string name</code> | <i>The name of the Image.</i> |
| <code>overprintGap</code> | <code>boolean overprintGap</code> | <i>If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors.</i> |
| <code>overprintStroke</code> | <code>boolean overprintStroke</code> | <i>If true, the Image's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors.</i> |
| <code>parentPage</code> | <code>{Page} *parentPage</code> | <i>The page on which this page item appears.</i> |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| <code>rightLineEnd</code> | <code>#ArrowHead rightLineEnd</code> | <i>The arrowhead applied to the end of the path.</i> |
| <code>strokeAlignment</code> | <code>#StrokeAlignment strokeAlignment</code> | <i>The stroke alignment applied to the Image.</i> |
| <code>strokeColor</code> | <code>variant strokeColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the Image. Type: Swatch or String.</i> |
| <code>strokeCornerAdjustment</code> | <code>#StrokeCornerAdjustment strokeCornerAdjustment</code> | <i>The corner adjustment applied to the Image.</i> |

| IMAGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|--|---|
| strokeDashAndGap | <code>variant strokeDashAndGap</code> | The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings). |
| strokeTint | <code>double strokeTint</code> | The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| strokeType | <code>variant strokeType</code> | The name of the stroke style to apply. Type: StrokeStyle or String. |
| strokeWeight | <code>variant strokeWeight</code> | The weight (in points) to apply to the Image's stroke. Type: Unit (Double or String). |
| timingSettings | <code>{TimingSetting} *timingSettings</code> | The object timing settings. |
| topLeftCornerOption | <code>#CornerOptions topLeftCornerOption</code> | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. |
| topLeftCornerRadius | <code>variant topLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String). |
| topRightCornerOption | <code>#CornerOptions topRightCornerOption</code> | The shape to apply to the top right corner of rectangular shapes. |
| topRightCornerRadius | <code>variant topRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String). |
| visible | <code>boolean visible</code> | If true, the Image is visible. |

InsertionPoint

Object available in ID 3, 4, 5, 6, 7

| INSERTIONPOINT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|--|--|
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| changeTransliterate | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| findHyperlinks | <code>{Objects} findHyperlinks(variant SortOrder)</code> | Finds hyperlink sources that intersecting with specified text range. / Return value: The found hyperlinks. / SortOrder: The sort order of found ranges. Type: idRangeSortOrder enumerator. |
| findTransliterate | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| showText | <code>void showText()</code> | Jump to the text range. |

| INSERTIONPOINT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|--|
| <code>allowArbitraryHyphenation</code> | <code>boolean allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. |
| <code>autoTcy</code> | <code>const autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. |
| <code>autoTcyIncludeRoman</code> | <code>boolean autoTcyIncludeRoman</code> | If true, auto tcy includes Roman characters. |
| <code>bunriKinshi</code> | <code>boolean bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. |
| <code>characterAlignment</code> | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| <code>characterRotation</code> | <code>double characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. |
| <code>cjkGridTracking</code> | <code>boolean cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| <code>glyphForm</code> | <code>#AlternateGlyphForms glyphForm</code> | The glyph variant to substitute for standard glyphs. |
| <code>gridAlignment</code> | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| <code>gridGyoudori</code> | <code>const gridGyoudori</code> | The manual gyoudori setting. |
| <code>jidori</code> | <code>const jidori</code> | The number of grid squares in which to arrange the text. |
| <code>keepWithPrevious</code> | <code>boolean keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. |
| <code>kentenAlignment</code> | <code>#KntenAlignment kentenAlignment</code> | The alignment of kenten characters relative to the parent characters. |
| <code>kentenCharacterSet</code> | <code>#KntenCharacterSet kentenCharacterSet</code> | The character set used for the custom kenten character. Note: Valid only when kenten kind is custom. |
| <code>kentenCustomCharacter</code> | <code>string kentenCustomCharacter</code> | The character used for kenten. Note: Valid only when kenten kind is custom. |
| <code>kentenFillColor</code> | <code>variant kentenFillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch or String. |
| <code>kentenFont</code> | <code>variant kentenFont</code> | The font to use for kenten characters. Type: Font or String. |
| <code>kentenFontSize</code> | <code>double kentenFontSize</code> | The size (in points) of kenten characters. |
| <code>kentenFontStyle</code> | <code>variant kentenFontStyle</code> | The font style of kenten characters. Type: String or idNothingEnum enumerator. |
| <code>kentenKind</code> | <code>#KntenCharacter kentenKind</code> | The style of kenten characters. |
| <code>kentenOverprintFill</code> | <code>#AdornmentOverprint kentenOverprintFill</code> | The method of overprinting the kenten fill. |
| <code>kentenOverprintStroke</code> | <code>#AdornmentOverprint kentenOverprintStroke</code> | The method of overprinting the kenten stroke. |
| <code>kentenPlacement</code> | <code>double kentenPlacement</code> | The distance between kenten characters and their parent characters. |
| <code>kentenPosition</code> | <code>#RubyKntenPosition kentenPosition</code> | The kenten position relative to the parent character. |
| <code>kentenStrokeColor</code> | <code>variant kentenStrokeColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of kenten characters. Type: Swatch or String. |
| <code>kentenStrokeTint</code> | <code>double kentenStrokeTint</code> | The stroke tint (as a percentage) of kenten characters. (Range: 0 to 100). |
| <code>kentenTint</code> | <code>double kentenTint</code> | The fill tint (as a percentage) of kenten characters. (Range: 0 to 100). |
| <code>kentenWeight</code> | <code>double kentenWeight</code> | The stroke weight (in points) of kenten characters. |
| <code>kentenXScale</code> | <code>double kentenXScale</code> | The horizontal size of kenten characters as a percent of the original size. |

| INSERTIONPOINT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|---|--|
| kentenYScale | <code>double kentenYScale</code> | The vertical size of kenten characters as a percent of the original size. |
| kinsokuHangType | <code>#KinsokuHangTypes kinsokuHangType</code> | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. |
| kinsokuSet | <code>variant kinsokuSet</code> | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator or String. |
| kinsokuType | <code>#KinsokuType kinsokuType</code> | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. |
| leadingAki | <code>double leadingAki</code> | The amount of space before each character. |
| leadingModel | <code>#LeadingModel leadingModel</code> | The point from which leading is measured from line to line. |
| mojikumi | <code>variant mojikumi</code> | The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String or idMojikumiTableDefaults enumerator. |
| multiStateObjects | <code>{MultiStateObjects} *multiStateObjects</code> | A collection of multi-state objects. |
| otfHVKana | <code>boolean otfHVKana</code> | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. |
| otfProportionalMetrics | <code>boolean otfProportionalMetrics</code> | If true, kerns according to proportional CJK metrics in OpenType fonts. |
| otfRomanItalics | <code>boolean otfRomanItalics</code> | If true, applies italics to half-width alphanumerics. |
| paragraphGyoudori | <code>boolean paragraphGyoudori</code> | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. |
| preferences | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| rensuuji | <code>boolean rensuuji</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. |
| rotateSingleByteCharacters | <code>boolean rotateSingleByteCharacters</code> | If true, rotates Roman characters in vertical text. |
| rubyAlignment | <code>#RubyAlignments rubyAlignment</code> | The ruby alignment. |
| rubyAutoAlign | <code>boolean rubyAutoAlign</code> | If true, auto aligns ruby. |
| rubyAutoScaling | <code>boolean rubyAutoScaling</code> | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. |
| rubyAutoTcyAutoSize | <code>boolean rubyAutoTcyAutoSize</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. |
| rubyAutoTcyDigits | <code>const rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. |
| rubyAutoTcyIncludeRoman | <code>boolean rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. |
| rubyFill | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch or String. |
| rubyFlag | <code>boolean rubyFlag</code> | If true, ruby is on. |
| rubyFont | <code>variant rubyFont</code> | The font applied to ruby characters. Type: Font or String. |
| rubyFontSize | <code>double rubyFontSize</code> | The size (in points) of ruby characters. |
| rubyFontStyle | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: String or idNothingEnum enumerator. |
| rubyOpenTypePro | <code>boolean rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. |
| rubyOverhang | <code>boolean rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. |

| INSERTIONPOINT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|--|---|
| <code>rubyOverprintFill</code> | <code>#AdornmentOverprint rubyOverprintFill</code> | <i>The method of overprinting the ruby fill.</i> |
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | <i>The method of overprinting the ruby stroke.</i> |
| <code>rubyParentOverhangAmount</code> | <code>#RubyOverhang rubyParentOverhangAmount</code> | <i>The amount by which ruby characters can overhang the parent text.</i> |
| <code>rubyParentScalingPercent</code> | <code>double rubyParentScalingPercent</code> | <i>The amount (as a percentage) to scale the parent text size to determine the ruby text size.</i> |
| <code>rubyParentSpacing</code> | <code>#RubyParentSpacing rubyParentSpacing</code> | <i>The ruby spacing relative to the parent text.</i> |
| <code>rubyPosition</code> | <code>#RubyKentePosition rubyPosition</code> | <i>The position of ruby characters relative to the parent text.</i> |
| <code>rubyString</code> | <code>string rubyString</code> | <i>The ruby string contents.</i> |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String.</i> |
| <code>rubyStrokeTint</code> | <code>double rubyStrokeTint</code> | <i>The stroke tint (as a percentage) of ruby characters.</i> |
| <code>rubyTint</code> | <code>double rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100).</i> |
| <code>rubyType</code> | <code>#RubyTypes rubyType</code> | <i>The ruby type.</i> |
| <code>rubyWeight</code> | <code>double rubyWeight</code> | <i>The stroke weight (in points) of ruby characters.</i> |
| <code>rubyXOffset</code> | <code>double rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters.</i> |
| <code>rubyXScale</code> | <code>double rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size.</i> |
| <code>rubyYOffset</code> | <code>double rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters.</i> |
| <code>rubyYScale</code> | <code>double rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size.</i> |
| <code>scaleAffectsLineHeight</code> | <code>boolean scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled.</i> |
| <code>shataiAdjustRotation</code> | <code>boolean shataiAdjustRotation</code> | <i>If true, applies shatai rotation.</i> |
| <code>shataiAdjustTsume</code> | <code>boolean shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume.</i> |
| <code>shataiDegreeAngle</code> | <code>double shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees).</i> |
| <code>shataiMagnification</code> | <code>double shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply.</i> |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | <i>The minimum space after a span or a split column. Type: Unit (Double or String).</i> |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | <i>The minimum space before a span or a split column. Type: Unit (Double or String).</i> |
| <code>spanColumnType</code> | <code>#SpanColumnTypeOptions spanColumnType</code> | <i>Whether a paragraph should be a single column, span columns or split columns.</i> |
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator.</i> |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | <i>The inside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>tatechuyoko</code> | <code>boolean tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text.</i> |

| INSERTIONPOINT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--------------------------------------|---|
| tatechuyokoXOffset | double tatechuyokoXOffset | The horizontal offset for horizontal characters in vertical text. |
| tatechuyokoYOffset | double tatechuyokoYOffset | The vertical offset for horizontal characters in vertical text. |
| trailingAki | double trailingAki | The amount of space after each character. |
| treatIdeographicSpaceAsSpace | boolean treatIdeographicSpaceAsSpace | If true, ideographic spaces will not wrap to the next line like text characters. |
| tsume | double tsume | The amount of horizontal character compression. |
| warichu | boolean warichu | If true, turns on warichu. |
| warichuAlignment | #WarichuAlignment warichuAlignment | The warichu alignment. |
| warichuCharsAfterBreak | const warichuCharsAfterBreak | The minimum number of characters allowed after a line break. |
| warichuCharsBeforeBreak | const warichuCharsBeforeBreak | The minimum number of characters allowed before a line break. |
| warichuLines | const warichuLines | The number of lines of warichu within a single normal line. |
| warichuLineSpacing | double warichuLineSpacing | The gap between lines of warichu characters. |
| warichuSize | double warichuSize | The amount (as a percentage) to scale parent text size to determine warichu size. |

JPEGExportPreference

Object available in ID 3, 4, 5, 6, 7

| JPEGEXPORTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|------------------------------------|---|
| antiAlias | boolean antiAlias | If true, use anti-aliasing for text and vectors during export. |
| embedColorProfile | boolean embedColorProfile | True to embed the color profile, false otherwise. |
| exportResolution | double exportResolution | The export resolution expressed as a real number instead of an integer. (Range: 1.0 to 2400.0). Type: Double (1 - 2400). |
| jpegColorSpace | #JpegColorSpaceEnum jpegColorSpace | One of RGB, CMYK or Gray. |
| simulateOverprint | boolean simulateOverprint | If true, simulates the effects of overprinting spot and process colors in the same way they would occur when printing. |
| useDocumentBleeds | boolean useDocumentBleeds | If true, uses the document's bleed settings in the exported JPEG. |

Language

Object available in ID 3, 4, 5, 6, 7

| LANGUAGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------|-----------------------|--|
| icuLocaleName | string *icuLocaleName | The full name of the Language object's ICU locale. |

LanguageWithVendors

Object available in ID 3, 4, 5, 6, 7

| LANGUAGEWITHVENDORS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|------------------------------------|--|
| <code>icuLocaleName</code> | <code>string *icuLocaleName</code> | The full name of the Language object's ICU locale. |

Layer

Object available in ID 3, 4, 5, 6, 7

| LAYER CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|---|--------------------------------------|
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | <code>EPSTexts</code> . |
| <code>multiStateObjects</code> | <code>{MultiStateObjects} *multiStateObjects</code> | A collection of multi-state objects. |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | The spline items collection. |

Line

Object available in ID 3, 4, 5, 6, 7

| LINE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: <code>idExportFormat</code> enumerator or <code>String</code> . / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: <code>PDFExportPreset</code> . / VersionComments: The comment for this version. Type: <code>String</code> . / ForceSave: If true, forcibly saves a version. |
| <code>changeTransliterate</code> | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: <code>Boolean</code> . |
| <code>findHyperlinks</code> | <code>{Objects} findHyperlinks(variant SortOrder)</code> | Finds hyperlink sources that intersecting with specified text range. / Return value: The found hyperlinks. / SortOrder: The sort order of found ranges. Type: <code>idRangeSortOrder</code> enumerator. |
| <code>findTransliterate</code> | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: <code>Boolean</code> . |
| <code>showText</code> | <code>void showText()</code> | Jump to the text range. |
| <code>allowArbitraryHyphenation</code> | <code>boolean allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. |
| <code>autoTcy</code> | <code>const autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. |
| <code>autoTcyIncludeRoman</code> | <code>boolean autoTcyIncludeRoman</code> | If true, auto tcy includes Roman characters. |

| LINE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------|--|---|
| bunriKinshi | <code>boolean bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (-) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. |
| characterAlignment | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| characterRotation | <code>double characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. |
| cjkGridTracking | <code>boolean cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. |
| epsTexts | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| glyphForm | <code>#AlternateGlyphForms glyphForm</code> | The glyph variant to substitute for standard glyphs. |
| gridAlignment | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| gridGyoudori | <code>const gridGyoudori</code> | The manual gyoudori setting. |
| jidori | <code>const jidori</code> | The number of grid squares in which to arrange the text. |
| keepWithPrevious | <code>boolean keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. |
| kentenAlignment | <code>#KetenAlignment kentenAlignment</code> | The alignment of kenten characters relative to the parent characters. |
| kentenCharacterSet | <code>#KetenCharacterSet kentenCharacterSet</code> | The character set used for the custom kenten character. Note: Valid only when kenten kind is custom. |
| kentenCustomCharacter | <code>string kentenCustomCharacter</code> | The character used for kenten. Note: Valid only when kenten kind is custom. |
| kentenFillColor | <code>variant kentenFillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch or String. |
| kentenFont | <code>variant kentenFont</code> | The font to use for kenten characters. Type: Font or String. |
| kentenFontSize | <code>double kentenFontSize</code> | The size (in points) of kenten characters. |
| kentenFontStyle | <code>variant kentenFontStyle</code> | The font style of kenten characters. Type: String or idNothingEnum enumerator. |
| kentenKind | <code>#KetenCharacter kentenKind</code> | The style of kenten characters. |
| kentenOverprintFill | <code>#AdornmentOverprint kentenOverprintFill</code> | The method of overprinting the kenten fill. |
| kentenOverprintStroke | <code>#AdornmentOverprint kentenOverprintStroke</code> | The method of overprinting the kenten stroke. |
| kentenPlacement | <code>double kentenPlacement</code> | The distance between kenten characters and their parent characters. |
| kentenPosition | <code>#RubyKetenPosition kentenPosition</code> | The kenten position relative to the parent character. |
| kentenStrokeColor | <code>variant kentenStrokeColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of kenten characters. Type: Swatch or String. |
| kentenStrokeTint | <code>double kentenStrokeTint</code> | The stroke tint (as a percentage) of kenten characters. (Range: 0 to 100). |
| kentenTint | <code>double kentenTint</code> | The fill tint (as a percentage) of kenten characters. (Range: 0 to 100). |
| kentenWeight | <code>double kentenWeight</code> | The stroke weight (in points) of kenten characters. |
| kentenXScale | <code>double kentenXScale</code> | The horizontal size of kenten characters as a percent of the original size. |
| kentenYScale | <code>double kentenYScale</code> | The vertical size of kenten characters as a percent of the original size. |
| kinsokuHangType | <code>#KinsokuHangTypes kinsokuHangType</code> | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. |
| kinsokuSet | <code>variant kinsokuSet</code> | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator or String. |

| LINE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| <code>kinsokuType</code> | <code>#KinsokuType kinsokuType</code> | <i>The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined.</i> |
| <code>leadingAki</code> | <code>double leadingAki</code> | <i>The amount of space before each character.</i> |
| <code>leadingModel</code> | <code>#LeadingModel leadingModel</code> | <i>The point from which leading is measured from line to line.</i> |
| <code>mojikumi</code> | <code>variant mojikumi</code> | <i>The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String or idMojikumiTableDefaults enumerator.</i> |
| <code>multiStateObjects</code> | <code>{MultiStateObjects} *multiStateObjects</code> | <i>A collection of multi-state objects.</i> |
| <code>otfHVKana</code> | <code>boolean otfHVKana</code> | <i>If true, switches hiragana fonts, which have different glyphs for horizontal and vertical.</i> |
| <code>otfProportionalMetrics</code> | <code>boolean otfProportionalMetrics</code> | <i>If true, kerns according to proportional CJK metrics in OpenType fonts.</i> |
| <code>otfRomanItalics</code> | <code>boolean otfRomanItalics</code> | <i>If true, applies italics to half-width alphanumerics.</i> |
| <code>paragraphGyoudori</code> | <code>boolean paragraphGyoudori</code> | <i>If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph.</i> |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| <code>rensuuji</code> | <code>boolean rensuuji</code> | <i>If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers.</i> |
| <code>rotateSingleByteCharacters</code> | <code>boolean rotateSingleByteCharacters</code> | <i>If true, rotates Roman characters in vertical text.</i> |
| <code>rubyAlignment</code> | <code>#RubyAlignments rubyAlignment</code> | <i>The ruby alignment.</i> |
| <code>rubyAutoAlign</code> | <code>boolean rubyAutoAlign</code> | <i>If true, auto aligns ruby.</i> |
| <code>rubyAutoScaling</code> | <code>boolean rubyAutoScaling</code> | <i>If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent.</i> |
| <code>rubyAutoTcyAutoScale</code> | <code>boolean rubyAutoTcyAutoScale</code> | <i>If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em.</i> |
| <code>rubyAutoTcyDigits</code> | <code>const rubyAutoTcyDigits</code> | <i>The number of digits included in auto tcy (tate-chuu-yoko) in ruby.</i> |
| <code>rubyAutoTcyIncludeRoman</code> | <code>boolean rubyAutoTcyIncludeRoman</code> | <i>If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby.</i> |
| <code>rubyFill</code> | <code>variant rubyFill</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch or String.</i> |
| <code>rubyFlag</code> | <code>boolean rubyFlag</code> | <i>If true, ruby is on.</i> |
| <code>rubyFont</code> | <code>variant rubyFont</code> | <i>The font applied to ruby characters. Type: Font or String.</i> |
| <code>rubyFontSize</code> | <code>double rubyFontSize</code> | <i>The size (in points) of ruby characters.</i> |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | <i>The font style of ruby characters. Type: String or idNothingEnum enumerator.</i> |
| <code>rubyOpenTypePro</code> | <code>boolean rubyOpenTypePro</code> | <i>If true, uses OpenType Pro fonts for ruby.</i> |
| <code>rubyOverhang</code> | <code>boolean rubyOverhang</code> | <i>If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount.</i> |
| <code>rubyOverprintFill</code> | <code>#AdornmentOverprint rubyOverprintFill</code> | <i>The method of overprinting the ruby fill.</i> |
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | <i>The method of overprinting the ruby stroke.</i> |
| <code>rubyParentOverhangAmount</code> | <code>#RubyOverhang rubyParentOverhangAmount</code> | <i>The amount by which ruby characters can overhang the parent text.</i> |
| <code>rubyParentScalingPercent</code> | <code>double rubyParentScalingPercent</code> | <i>The amount (as a percentage) to scale the parent text size to determine the ruby text size.</i> |

| LINE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|--|
| rubyParentSpacing | <code>#RubyParentSpacing</code> rubyParentSpacing | The ruby spacing relative to the parent text. |
| rubyPosition | <code>#RubyKetenPosition</code> rubyPosition | The position of ruby characters relative to the parent text. |
| rubyString | <code>string</code> rubyString | The ruby string contents. |
| rubyStroke | <code>variant</code> rubyStroke | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String. |
| rubyStrokeTint | <code>double</code> rubyStrokeTint | The stroke tint (as a percentage) of ruby characters. |
| rubyTint | <code>double</code> rubyTint | The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). |
| rubyType | <code>#RubyTypes</code> rubyType | The ruby type. |
| rubyWeight | <code>double</code> rubyWeight | The stroke weight (in points) of ruby characters. |
| rubyXOffset | <code>double</code> rubyXOffset | The amount of horizontal space between ruby and parent characters. |
| rubyXScale | <code>double</code> rubyXScale | The horizontal size of ruby characters, specified as a percent of the original size. |
| rubyYOffset | <code>double</code> rubyYOffset | The amount of vertical space between ruby and parent characters. |
| rubyYScale | <code>double</code> rubyYScale | The vertical size of ruby characters, specified as a percent of the original size. |
| scaleAffectsLineHeight | <code>boolean</code> scaleAffectsLineHeight | If true, the line changes size when characters are scaled. |
| shataiAdjustRotation | <code>boolean</code> shataiAdjustRotation | If true, applies shatai rotation. |
| shataiAdjustTsume | <code>boolean</code> shataiAdjustTsume | If true, adjusts shatai tsume. |
| shataiDegreeAngle | <code>double</code> shataiDegreeAngle | The shatai lens angle (in degrees). |
| shataiMagnification | <code>double</code> shataiMagnification | The amount (as a percentage) of shatai obliquing to apply. |
| spanColumnMinSpaceAfter | <code>variant</code> spanColumnMinSpaceAfter | The minimum space after a span or a split column. Type: Unit (Double or String). |
| spanColumnMinSpaceBefore | <code>variant</code> spanColumnMinSpaceBefore | The minimum space before a span or a split column. Type: Unit (Double or String). |
| spanColumnType | <code>#SpanColumnTypeOptions</code> spanColumnType | Whether a paragraph should be a single column, span columns or split columns. |
| spanSplitColumnCount | <code>variant</code> spanSplitColumnCount | The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator. |
| splineItems | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| splitColumnInsideGutter | <code>variant</code> splitColumnInsideGutter | The inside gutter if the paragraph splits columns. Type: Unit (Double or String). |
| splitColumnOutsideGutter | <code>variant</code> splitColumnOutsideGutter | The outside gutter if the paragraph splits columns. Type: Unit (Double or String). |
| tatechuyoko | <code>boolean</code> tatechuyoko | If true, makes the character horizontal in vertical text. |
| tatechuyokoXOffset | <code>double</code> tatechuyokoXOffset | The horizontal offset for horizontal characters in vertical text. |
| tatechuyokoYOffset | <code>double</code> tatechuyokoYOffset | The vertical offset for horizontal characters in vertical text. |
| trailingAki | <code>double</code> trailingAki | The amount of space after each character. |
| treatIdeographicSpaceAsSpace | <code>boolean</code> treatIdeographicSpaceAsSpace | If true, ideographic spaces will not wrap to the next line like text characters. |
| tsume | <code>double</code> tsume | The amount of horizontal character compression. |

| LINE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|---|---|
| <code>warichu</code> | <code>boolean warichu</code> | If true, turns on warichu. |
| <code>warichuAlignment</code> | <code>#WarichuAlignment warichuAlignment</code> | The warichu alignment. |
| <code>warichuCharsAfterBreak</code> | <code>const warichuCharsAfterBreak</code> | The minimum number of characters allowed after a line break. |
| <code>warichuCharsBeforeBreak</code> | <code>const warichuCharsBeforeBreak</code> | The minimum number of characters allowed before a line break. |
| <code>warichuLines</code> | <code>const warichuLines</code> | The number of lines of warichu within a single normal line. |
| <code>warichuLineSpacing</code> | <code>double warichuLineSpacing</code> | The gap between lines of warichu characters. |
| <code>warichuSize</code> | <code>double warichuSize</code> | The amount (as a percentage) to scale parent text size to determine warichu size. |

Link

Object available in ID 3, 4, 5, 6, 7

| LINK CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|---|--------------------------------------|
| <code>links</code> | <code>{Links} *links</code> | A collection of links. |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | A collection of preferences objects. |

MarginPreference

Object available in ID 3, 4, 5, 6, 7

| MARGINPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|--------------------------------------|
| <code>columnDirection</code> | <code>#HorizontalOrVertical columnDirection</code> | The direction of text in the column. |

MasterSpread

Object available in ID 3, 4, 5, 6, 7

| MASTERSPREAD CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|---|---|
| <code>resolve</code> | <code>variant resolve(variant Location, #CoordinateSpaces In, boolean ConsideringRulerUnits)</code> | Get the coordinates of the given location in the specified coordinate system. / Return value: The resolved location for each of the targeted page items. Type: Array of 2 Doubles or Array of Arrays of 2 Doubles. / Location: The location requested. Type: Array of 2 Doubles, idAnchorPoint enumerators, or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / In: The coordinate space to use. / ConsideringRulerUnits: If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| <code>select</code> | <code>void select(#SelectionOptions ExistingSelection)</code> | Selects the object. / ExistingSelection: The selection status of the MasterSpread in relation to previously selected objects. |

| MASTERSPREAD CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|--|--|
| transform | <code>void transform(#CoordinateSpaces In, variant From, variant WithMatrix, variant ReplacingCurrent, boolean ConsideringRulerUnits)</code> | Transform the page item. / In: The coordinate space to use. / From: The temporary origin during the transformation. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / WithMatrix: Transform matrix. Type: Array of 6 Doubles or TransformationMatrix. / ReplacingCurrent: Transform components to consider; providing this optional parameter causes the target's existing transform components to be replaced with new values. Without this parameter, the given matrix is concatenated onto the target's existing transform combining the effect of the two. Type: idMatrixContent enumerator, Array of idMatrixContent enumerators or Long Integer. / ConsideringRulerUnits: If true then a ruler based origin is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| transformValuesOf | <code>{Objects} transformValuesOf(#CoordinateSpaces In)</code> | Get the transformation values of the page item. / Return value: A list of the transformation matrices. / In: The coordinate space to use. |
| epsTexts | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| multiStateObjects | <code>{MultiStateObjects} *multiStateObjects</code> | A collection of multi-state objects. |
| pageColor | <code>variant pageColor</code> | The color label of the MasterSpread, specified either as an array of three doubles, each in the range 0 to 255 and representing R, G, and B values, or as a UI color. Type: Array of 3 Doubles (0 - 255) or idUIColors enumerator or idPageColorOptions enumerator. |
| preferences | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| splineItems | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| timingSettings | <code>{TimingSetting} *timingSettings</code> | The object timing settings. |

Movie

Object available in ID 3, 4, 5, 6, 7

| MOVIE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|--|--|
| applyObjectType | <code>void applyObjectType(ObjectStyle Using, boolean ClearingOverrides, boolean ClearingOverridesThroughRootObjectType)</code> | Applies the specified object style. / Using: The object style to apply. / ClearingOverrides: If true, clears the Movie's existing attributes before applying the style. / ClearingOverridesThroughRootObjectType: If true, clears attributes and formatting applied to the Movie that are not defined in the object style. |
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| changeObject | <code>{Objects} changeObject(variant ReverseOrder)</code> | Finds objects that match the find what value and replace the objects with the change to value. / Return value: The replaced object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |

| MOVIE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| clearObjectStyleOverrides | <code>void clearObjectStyleOverrides()</code> | Clear overrides for object style. |
| export | <code>void export(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | Exports the object(s) to a file. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| findObject | <code>{Objects} findObject(variant ReverseOrder)</code> | Finds objects that match the find what value. / Return value: The found object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| flipItem | <code>void flipItem(#Flip Given, variant Around)</code> | Flips the Movie. / Given: The axis around which to flip the Movie. / Around: The point around which to flip the Movie. Type: Array of 2 Units (Doubles or Strings) or idAnchorPoint enumerator. |
| place | <code>variant place(string FileName, boolean ShowingOptions)</code> | Places the file. / Return value: The placed file. Type: Array of PageItems, Graphics, Movies, Sounds, Stories or Texts. / FileName: The file to place. / ShowingOptions: Whether to display the import options dialog. |
| reframe | <code>void reframe(variant In, variant OpposingCorners)</code> | Move the bounding box of the page item. / In: The bounding box to resize. Type: idCoordinateSpaces enumerator or Ordered array containing CoordinateSpace:idCoordinateSpaces enumerator, BoundsKind:idBoundingBoxLimits enumerator. / OpposingCorners: Opposing corners of new bounding box in the given coordinate space. Type: Array of 2 Arrays of 2 Doubles. |
| resolve | <code>variant resolve(variant Location, #CoordinateSpaces In, boolean ConsideringRulerUnits)</code> | Get the coordinates of the given location in the specified coordinate system. / Return value: The resolved location for each of the targeted page items. Type: Array of 2 Doubles or Array of Arrays of 2 Doubles. / Location: The location requested. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / In: The coordinate space to use. / ConsideringRulerUnits: If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| transformAgain | <code>variant transformAgain()</code> | Transforms the Movie using the last transformation performed on any object. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformAgainIndividually | <code>variant transformAgainIndividually()</code> | Transforms the Movie using the last transformation performed on any Movie. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformSequenceAgain | <code>variant transformSequenceAgain()</code> | Transforms the Movie using the last sequence of transform operations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformSequenceAgainIndividually | <code>variant transformSequenceAgainIndividually()</code> | Transforms the Movie using the last sequence of transformations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformValuesOf | <code>{Objects} transformValuesOf(#CoordinateSpaces In)</code> | Get the transformation values of the page item. / Return value: A list of the transformation matrices. / In: The coordinate space to use. |
| absoluteFlip | <code>#Flip absoluteFlip</code> | Indicates whether the Movie has been flipped independently of its parent object and, if yes, the direction in which the Movie was flipped. |

| MOVIE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|---|
| <code>absoluteHorizontalScale</code> | <code>double absoluteHorizontalScale</code> | <i>The horizontal scale of the Movie relative to its containing object.</i> |
| <code>absoluteRotationAngle</code> | <code>double absoluteRotationAngle</code> | <i>The rotation angle of the Movie relative to its containing object. (Range: -360 to 360).</i> |
| <code>absoluteShearAngle</code> | <code>double absoluteShearAngle</code> | <i>The skewing angle of the Movie relative to its containing object. (Range: -360 to 360).</i> |
| <code>absoluteVerticalScale</code> | <code>double absoluteVerticalScale</code> | <i>The vertical scale of the Movie relative to its containing object.</i> |
| <code>allGraphics</code> | <code>{Objects} *allGraphics</code> | <i>Lists all graphics contained by the Movie.</i> |
| <code>allPageItems</code> | <code>{Objects} *allPageItems</code> | <i>Lists all page items contained by the Movie.</i> |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | <i>The page item animation settings.</i> |
| <code>appliedObjectStyle</code> | <code>{ObjectStyle} appliedObjectStyle</code> | <i>The object style applied to the Movie.</i> |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>contentTransparencySettings</code> | <code>{ContentTransparencySetting} *contentTransparencySettings</code> | <i>Transparency settings for the content of the Movie.</i> |
| <code>controllerSkin</code> | <code>string controllerSkin</code> | <i>The video controller skin name.</i> |
| <code>endCap</code> | <code>#EndCap endCap</code> | <i>The end shape of an open path.</i> |
| <code>endJoin</code> | <code>#EndJoin endJoin</code> | <i>The corner join applied to the Movie.</i> |
| <code>fillColor</code> | <code>variant fillColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of the Movie. . Type: Swatch or String.</i> |
| <code>fillTint</code> | <code>double fillTint</code> | <i>The percent of tint to use in the Movie's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>fillTransparencySettings</code> | <code>{FillTransparencySetting} *fillTransparencySettings</code> | <i>Transparency settings for the fill applied to the Movie.</i> |
| <code>flip</code> | <code>#Flip flip</code> | <i>The direction in which to flip the printed image.</i> |
| <code>gapColor</code> | <code>{Swatch} gapColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type.</i> |
| <code>gapTint</code> | <code>double gapTint</code> | <i>The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>gradientFillAngle</code> | <code>double gradientFillAngle</code> | <i>The angle of a linear gradient applied to the fill of the Movie. (Range: -180 to 180).</i> |
| <code>gradientFillLength</code> | <code>variant gradientFillLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the Movie. Type: Unit (Double or String).</i> |
| <code>gradientFillStart</code> | <code>variant gradientFillStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the fill of the Movie, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>gradientStrokeAngle</code> | <code>double gradientStrokeAngle</code> | <i>The angle of a linear gradient applied to the stroke of the Movie. (Range: -180 to 180).</i> |

| MOVIE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|---|--|
| gradientStrokeLength | variant gradientStrokeLength | The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the Movie. Type: Unit (Double or String). |
| gradientStrokeStart | variant gradientStrokeStart | The starting point (in page coordinates) of a gradient applied to the stroke of the Movie, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| graphics | {Graphics} •graphics | A collection of imported graphics in any graphic file format (vector, metafile, or bitmap). |
| leftLineEnd | #ArrowHead leftLineEnd | The arrowhead applied to the start of the path. |
| localDisplaySetting | #DisplaySettingOptions localDisplaySetting | Display performance options for the Movie. |
| miterLimit | double miterLimit | The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500). |
| movieLoop | boolean movieLoop | If true, movie loops forever. |
| navigationPoints | {NavigationPoints} •navigationPoints | A collection of navigation points. |
| nonprinting | boolean nonprinting | If true, the Movie does not print. |
| overprintFill | boolean overprintFill | If true, the Movie's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors. |
| overprintGap | boolean overprintGap | If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors. |
| overprintStroke | boolean overprintStroke | If true, the Movie's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors. |
| pageItems | {PageItems} •pageItems | The page items collection, which can be used to process all page items in a container (such as a document, page, or group), regardless of type. |
| parentPage | {Page} •parentPage | The page on which this page item appears. |
| paths | {Paths} •paths | A collection of paths. |
| preferences | {Preferences} •preferences | A collection of preferences objects. |
| rightLineEnd | #ArrowHead rightLineEnd | The arrowhead applied to the end of the path. |
| rotationAngle | double rotationAngle | The rotation angle of the Movie. (Range: -360 to 360). |
| shearAngle | double shearAngle | The skewing angle applied to the Movie. (Range: -360 to 360). |
| showController | boolean showController | If true, displays controller skin with mouse rollover. |
| strokeAlignment | #StrokeAlignment strokeAlignment | The stroke alignment applied to the Movie. |
| strokeColor | variant strokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the Movie. Type: Swatch or String. |
| strokeCornerAdjustment | #StrokeCornerAdjustment strokeCornerAdjustment | The corner adjustment applied to the Movie. |
| strokeDashAndGap | variant strokeDashAndGap | The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings). |
| strokeTint | double strokeTint | The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| strokeTransparencySettings | {StrokeTransparencySetting} •strokeTransparencySettings | Transparency settings for the stroke. |
| strokeType | variant strokeType | The name of the stroke style to apply. Type: StrokeStyle or String. |

| MOVIE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|--|
| <code>strokeWeight</code> | <code>variant strokeWeight</code> | <i>The weight (in points) to apply to the Movie's stroke. Type: Unit (Double or String).</i> |
| <code>textWrapPreferences</code> | <code>{TextWrapPreference} *textWrapPreferences</code> | <i>The text wrap preference properties that define the default formatting for wrapping text around objects.</i> |
| <code>timingSettings</code> | <code>{TimingSetting} *timingSettings</code> | <i>The object timing settings.</i> |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>transparencySettings</code> | <code>{TransparencySetting} *transparencySettings</code> | <i>Transparency settings.</i> |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the Movie is visible.</i> |

MovieBehavior

Object available in ID 3, 4, 5, 6, 7

| MOVIEBEHAVIOR CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|--|
| <code>navigationPointID</code> | <code>integer navigationPointID</code> | <i>The id of the navigation point to play from. This property is ignored for all operations other than Play From Navigation Point.</i> |

Oval

Object available in ID 3, 4, 5, 6, 7

| OVAL CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| <code>clearFrameFittingOptions</code> | <code>void clearFrameFittingOptions()</code> | <i>Removes the frame fittings options and resets it to the initial state.</i> |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | <i>The page item animation settings.</i> |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |

| OVAL CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|---|--|
| bottomLeftCornerRadius | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String).</i> |
| bottomRightCornerOption | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| bottomRightCornerRadius | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| buttons | <code>{Buttons} *buttons</code> | <i>A collection of buttons.</i> |
| epsTexts | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |
| formFields | <code>{FormFields} *formFields</code> | <i>A collection of form fields.</i> |
| mediaItems | <code>{MediaItems} *mediaItems</code> | <i>The media items collection.</i> |
| multiStateObjects | <code>{MultiStateObjects} *multiStateObjects</code> | <i>A collection of multi-state objects.</i> |
| name | <code>string name</code> | <i>The name of the Oval.</i> |
| parentPage | <code>{Page} *parentPage</code> | <i>The page on which this page item appears.</i> |
| preferences | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| splineItems | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |
| timingSettings | <code>{TimingSetting} *timingSettings</code> | <i>The object timing settings.</i> |
| topLeftCornerOption | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| topLeftCornerRadius | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| topRightCornerOption | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| topRightCornerRadius | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| visible | <code>boolean visible</code> | <i>If true, the Oval is visible.</i> |

Page

Object available in ID 3, 4, 5, 6, 7

| PAGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------|--|---|
| reframe | <code>void reframe(variant In, variant OpposingCorners)</code> | <i>Move the bounding box of the page item. / In: The bounding box to resize. Type: idCoordinateSpaces enumerator or Ordered array containing CoordinateSpace:idCoordinateSpaces enumerator, BoundsKind:idBoundingBoxLimits enumerator. / OpposingCorners: Opposing corners of new bounding box in the given coordinate space. Type: Array of 2 Arrays of 2 Doubles.</i> |

| PAGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|---|
| resize | <code>void resize(variant In, variant From, #ResizeMethods By, variant Values, boolean ResizeIndividually, boolean ConsideringRulerUnits)</code> | <i>Resize the page item.</i> / <i>In:</i> The bounding box to resize. Type: <code>idCoordinateSpaces</code> enumerator, <code>idBoundingBoxLimits</code> enumerator or Ordered array containing <code>CoordinateSpace</code> : <code>idCoordinateSpaces</code> enumerator, <code>BoundsKind</code> : <code>idBoundingBoxLimits</code> enumerator. / <i>From:</i> The transform origin. Legal specifications: relative to bounding box: <code>anchor</code> <code>{anchor {x,y}}</code> , <code>bounds kind [, coordinate space]}</code> ; relative to coordinate space: <code>{x,y}</code> <code>{x,y [, coordinate space]}</code> ; relative to layout window ruler: <code>{x,y}, page index</code> <code>bounds kind</code> . Type: Array of 2 Doubles, <code>idAnchorPoint</code> enumerator or Array of Arrays of 2 Doubles, <code>idCoordinateSpaces</code> enumerators, <code>idAnchorPoint</code> enumerators, <code>idBoundingBoxLimits</code> enumerators or Long Integers. / <i>By:</i> How the current dimensions are affected by the given values. / <i>Values:</i> The width and height values. Legal dimensions specifications: <code>{x, y [, coordinate space]}</code> , <code>{x, resize constraint [, coordinate space]}</code> , or <code>{resize constraint, y [, coordinate space]}</code> ; where x and y are real numbers and coordinate space is used to determine _only_ the unit of length for x and y; coordinate space is ignored for the 'current dimensions' resize method. Type: Array of Doubles, <code>idResizeConstraints</code> enumerators or <code>idCoordinateSpaces</code> enumerators. / <i>ResizeIndividually:</i> If false and multiple page items are targeted, the new dimensions are attained only by moving the individual items rather than resizing them. / <i>ConsideringRulerUnits:</i> If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| select | <code>void select(#SelectionOptions ExistingSelection)</code> | Selects the object. / <i>ExistingSelection:</i> The selection status of the Page in relation to previously selected objects. |
| transform | <code>void transform(#CoordinateSpaces In, variant From, variant WithMatrix, variant ReplacingCurrent, boolean ConsideringRulerUnits)</code> | <i>Transform the page item.</i> / <i>In:</i> The coordinate space to use. / <i>From:</i> The temporary origin during the transformation. Type: Array of 2 Doubles, <code>idAnchorPoint</code> enumerator or Array of Arrays of 2 Doubles, <code>idCoordinateSpaces</code> enumerators, <code>idAnchorPoint</code> enumerators, <code>idBoundingBoxLimits</code> enumerators or Long Integers. / <i>WithMatrix:</i> Transform matrix. Type: Array of 6 Doubles or <code>TransformationMatrix</code> . / <i>ReplacingCurrent:</i> Transform components to consider; providing this optional parameter causes the target's existing transform components to be replaced with new values. Without this parameter, the given matrix is concatenated onto the target's existing transform combining the effect of the two. Type: <code>idMatrixContent</code> enumerator, Array of <code>idMatrixContent</code> enumerators or Long Integer. / <i>ConsideringRulerUnits:</i> If true then a ruler based origin is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| epsTexts | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |
| gridData | <code>{GridDataInformation} *gridData</code> | Default grid properties. Note: Applies to named, layout, and frame (story) grids. |
| masterPageTransform | <code>{TransformationMatrix} masterPageTransform</code> | The transform applied to the master page before it is applied to Page. |
| multiStateObjects | <code>{MultiStateObjects} *multiStateObjects</code> | A collection of multi-state objects. |
| pageColor | <code>variant pageColor</code> | The color label of the Page, specified either as an array of three doubles, each in the range 0 to 255 and representing R, G, and B values, or as a UI color. Type: Array of 3 Doubles (0 - 255) or <code>idUIColors</code> enumerator or <code>idPageColorOptions</code> enumerator. |
| preferences | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| splineItems | <code>{SplineItems} *splineItems</code> | The spline items collection. |

PageItem

Object available in ID 3, 4, 5, 6, 7

| PAGEITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------|---|--|
| asynchronousExportFile | {BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave) | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| animationSettings | {AnimationSetting} *animationSettings | The page item animation settings. |
| bottomLeftCornerOption | #CornerOptions bottomLeftCornerOption | The shape to apply to the bottom left corner of rectangular shapes. |
| bottomLeftCornerRadius | variant bottomLeftCornerRadius | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| bottomRightCornerOption | #CornerOptions bottomRightCornerOption | The shape to apply to the bottom right corner of rectangular shapes. |
| bottomRightCornerRadius | variant bottomRightCornerRadius | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| name | string name | The name of the PageItem. |
| parentPage | {Page} *parentPage | The page on which this page item appears. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |
| timingSettings | {TimingSetting} *timingSettings | The object timing settings. |
| topLeftCornerOption | #CornerOptions topLeftCornerOption | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. |
| topLeftCornerRadius | variant topLeftCornerRadius | The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String). |
| topRightCornerOption | #CornerOptions topRightCornerOption | The shape to apply to the top right corner of rectangular shapes. |
| topRightCornerRadius | variant topRightCornerRadius | The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String). |
| visible | boolean visible | If true, the PageItem is visible. |

PageReference

Object available in ID 3, 4, 5, 6, 7

| PAGEREference CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|-------------|--|
| name | string name | The name of the PageReference; this is an alias to the PageReference's label property. |

Paragraph

Object available in ID 3, 4, 5, 6, 7

| PARAGRAPH CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|--|
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| changeTransliterate | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| findHyperlinks | <code>{Objects} findHyperlinks(variant SortOrder)</code> | Finds hyperlink sources that intersecting with specified text range. / Return value: The found hyperlinks. / SortOrder: The sort order of found ranges. Type: idRangeSortOrder enumerator. |
| findTransliterate | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| showText | <code>void showText()</code> | Jump to the text range. |
| allowArbitraryHyphenation | <code>boolean allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. |
| autoTcy | <code>const autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. |
| autoTcyIncludeRoman | <code>boolean autoTcyIncludeRoman</code> | If true, auto tcy includes Roman characters. |
| bunriKinshi | <code>boolean bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. |
| characterAlignment | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| characterRotation | <code>double characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. |
| cjkGridTracking | <code>boolean cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. |
| epsTexts | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| glyphForm | <code>#AlternateGlyphForms glyphForm</code> | The glyph variant to substitute for standard glyphs. |
| gridAlignment | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| gridGyoudori | <code>const gridGyoudori</code> | The manual gyoudori setting. |
| jidori | <code>const jidori</code> | The number of grid squares in which to arrange the text. |
| keepWithPrevious | <code>boolean keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. |
| kntenAlignment | <code>#KntenAlignment kntenAlignment</code> | The alignment of knten characters relative to the parent characters. |
| kntenCharacterSet | <code>#KntenCharacterSet kntenCharacterSet</code> | The character set used for the custom knten character. Note: Valid only when knten kind is custom. |
| kntenCustomCharacter | <code>string kntenCustomCharacter</code> | The character used for knten. Note: Valid only when knten kind is custom. |

| PARAGRAPH CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|---|---|
| kntenFillColor | <code>variant</code> kntenFillColor | The swatch (color, gradient, tint, or mixed ink) applied to the fill of knten characters. Type: Swatch or String. |
| kntenFont | <code>variant</code> kntenFont | The font to use for knten characters. Type: Font or String. |
| kntenFontSize | <code>double</code> kntenFontSize | The size (in points) of knten characters. |
| kntenFontStyle | <code>variant</code> kntenFontStyle | The font style of knten characters. Type: String or idNothingEnum enumerator. |
| kntenKind | <code>#KntenCharacter</code> kntenKind | The style of knten characters. |
| kntenOverprintFill | <code>#AdornmentOverprint</code> kntenOverprintFill | The method of overprinting the knten fill. |
| kntenOverprintStroke | <code>#AdornmentOverprint</code> kntenOverprintStroke | The method of overprinting the knten stroke. |
| kntenPlacement | <code>double</code> kntenPlacement | The distance between knten characters and their parent characters. |
| kntenPosition | <code>#RubyKetenPosition</code> kntenPosition | The keten position relative to the parent character. |
| kntenStrokeColor | <code>variant</code> kntenStrokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of knten characters. Type: Swatch or String. |
| kntenStrokeTint | <code>double</code> kntenStrokeTint | The stroke tint (as a percentage) of knten characters. (Range: 0 to 100). |
| kntenTint | <code>double</code> kntenTint | The fill tint (as a percentage) of knten characters. (Range: 0 to 100). |
| kntenWeight | <code>double</code> kntenWeight | The stroke weight (in points) of knten characters. |
| kntenXScale | <code>double</code> kntenXScale | The horizontal size of knten characters as a percent of the original size. |
| kntenYScale | <code>double</code> kntenYScale | The vertical size of knen characters as a percent of the original size. |
| kinsokuHangType | <code>#KinsokuHangTypes</code> kinsokuHangType | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. |
| kinsokuSet | <code>variant</code> kinsokuSet | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator or String. |
| kinsokuType | <code>#KinsokuType</code> kinsokuType | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. |
| leadingAki | <code>double</code> leadingAki | The amount of space before each character. |
| leadingModel | <code>#LeadingModel</code> leadingModel | The point from which leading is measured from line to line. |
| mojikumi | <code>variant</code> mojikumi | The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String or idMojikumiTableDefaults enumerator. |
| multiStateObjects | <code>{MultiStateObjects}</code> *multiStateObjects | A collection of multi-state objects. |
| otfHVKana | <code>boolean</code> otfHVKana | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. |
| otfProportionalMetrics | <code>boolean</code> otfProportionalMetrics | If true, kerns according to proportional CJK metrics in OpenType fonts. |
| otfRomanItalics | <code>boolean</code> otfRomanItalics | If true, applies italics to half-width alphanumerics. |
| paragraphGyoudori | <code>boolean</code> paragraphGyoudori | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. |
| preferences | <code>{Preferences}</code> *preferences | A collection of preferences objects. |
| rensuuji | <code>boolean</code> rensuuji | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. |
| rotateSingleByteCharacters | <code>boolean</code> rotateSingleByteCharacters | If true, rotates Roman characters in vertical text. |

| PARAGRAPH CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|--|---|
| <code>rubyAlignment</code> | <code>#RubyAlignments rubyAlignment</code> | <i>The ruby alignment.</i> |
| <code>rubyAutoAlign</code> | <code>boolean rubyAutoAlign</code> | <i>If true, auto aligns ruby.</i> |
| <code>rubyAutoScaling</code> | <code>boolean rubyAutoScaling</code> | <i>If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent.</i> |
| <code>rubyAutoTcyAutoScale</code> | <code>boolean rubyAutoTcyAutoScale</code> | <i>If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em.</i> |
| <code>rubyAutoTcyDigits</code> | <code>const rubyAutoTcyDigits</code> | <i>The number of digits included in auto tcy (tate-chuu-yoko) in ruby.</i> |
| <code>rubyAutoTcyIncludeRoman</code> | <code>boolean rubyAutoTcyIncludeRoman</code> | <i>If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby.</i> |
| <code>rubyFill</code> | <code>variant rubyFill</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch or String.</i> |
| <code>rubyFlag</code> | <code>boolean rubyFlag</code> | <i>If true, ruby is on.</i> |
| <code>rubyFont</code> | <code>variant rubyFont</code> | <i>The font applied to ruby characters. Type: Font or String.</i> |
| <code>rubyFontSize</code> | <code>double rubyFontSize</code> | <i>The size (in points) of ruby characters.</i> |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | <i>The font style of ruby characters. Type: String or idNothingEnum enumerator.</i> |
| <code>rubyOpenTypePro</code> | <code>boolean rubyOpenTypePro</code> | <i>If true, uses OpenType Pro fonts for ruby.</i> |
| <code>rubyOverhang</code> | <code>boolean rubyOverhang</code> | <i>If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount.</i> |
| <code>rubyOverprintFill</code> | <code>#AdornmentOverprint rubyOverprintFill</code> | <i>The method of overprinting the ruby fill.</i> |
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | <i>The method of overprinting the ruby stroke.</i> |
| <code>rubyParentOverhangAmount</code> | <code>#RubyOverhang rubyParentOverhangAmount</code> | <i>The amount by which ruby characters can overhang the parent text.</i> |
| <code>rubyParentScalingPercent</code> | <code>double rubyParentScalingPercent</code> | <i>The amount (as a percentage) to scale the parent text size to determine the ruby text size.</i> |
| <code>rubyParentSpacing</code> | <code>#RubyParentSpacing rubyParentSpacing</code> | <i>The ruby spacing relative to the parent text.</i> |
| <code>rubyPosition</code> | <code>#RubyKetenPosition rubyPosition</code> | <i>The position of ruby characters relative to the parent text.</i> |
| <code>rubyString</code> | <code>string rubyString</code> | <i>The ruby string contents.</i> |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String.</i> |
| <code>rubyStrokeTint</code> | <code>double rubyStrokeTint</code> | <i>The stroke tint (as a percentage) of ruby characters.</i> |
| <code>rubyTint</code> | <code>double rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100).</i> |
| <code>rubyType</code> | <code>#RubyTypes rubyType</code> | <i>The ruby type.</i> |
| <code>rubyWeight</code> | <code>double rubyWeight</code> | <i>The stroke weight (in points) of ruby characters.</i> |
| <code>rubyXOffset</code> | <code>double rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters.</i> |
| <code>rubyXScale</code> | <code>double rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size.</i> |
| <code>rubyYOffset</code> | <code>double rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters.</i> |
| <code>rubyYScale</code> | <code>double rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size.</i> |
| <code>scaleAffectsLineHeight</code> | <code>boolean scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled.</i> |

| PARAGRAPH CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>shataiAdjustRotation</code> | <code>boolean shataiAdjustRotation</code> | If true, applies shatai rotation. |
| <code>shataiAdjustTsume</code> | <code>boolean shataiAdjustTsume</code> | If true, adjusts shatai tsume. |
| <code>shataiDegreeAngle</code> | <code>double shataiDegreeAngle</code> | The shatai lens angle (in degrees). |
| <code>shataiMagnification</code> | <code>double shataiMagnification</code> | The amount (as a percentage) of shatai obliquing to apply. |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | The minimum space after a span or a split column. Type: Unit (Double or String). |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | The minimum space before a span or a split column. Type: Unit (Double or String). |
| <code>spanColumnType</code> | <code>#SpanColumnTypeOptions spanColumnType</code> | Whether a paragraph should be a single column, span columns or split columns. |
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator. |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | The inside gutter if the paragraph splits columns. Type: Unit (Double or String). |
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | The outside gutter if the paragraph splits columns. Type: Unit (Double or String). |
| <code>tatechuyoko</code> | <code>boolean tatechuyoko</code> | If true, makes the character horizontal in vertical text. |
| <code>tatechuyokoXOffset</code> | <code>double tatechuyokoXOffset</code> | The horizontal offset for horizontal characters in vertical text. |
| <code>tatechuyokoYOffset</code> | <code>double tatechuyokoYOffset</code> | The vertical offset for horizontal characters in vertical text. |
| <code>trailingAki</code> | <code>double trailingAki</code> | The amount of space after each character. |
| <code>treatIdeographicSpaceAsSpace</code> | <code>boolean treatIdeographicSpaceAsSpace</code> | If true, ideographic spaces will not wrap to the next line like text characters. |
| <code>tsume</code> | <code>double tsume</code> | The amount of horizontal character compression. |
| <code>warichu</code> | <code>boolean warichu</code> | If true, turns on warichu. |
| <code>warichuAlignment</code> | <code>#WarichuAlignment warichuAlignment</code> | The warichu alignment. |
| <code>warichuCharsAfterBreak</code> | <code>const warichuCharsAfterBreak</code> | The minimum number of characters allowed after a line break. |
| <code>warichuCharsBeforeBreak</code> | <code>const warichuCharsBeforeBreak</code> | The minimum number of characters allowed before a line break. |
| <code>warichuLines</code> | <code>const warichuLines</code> | The number of lines of warichu within a single normal line. |
| <code>warichuLineSpacing</code> | <code>double warichuLineSpacing</code> | The gap between lines of warichu characters. |
| <code>warichuSize</code> | <code>double warichuSize</code> | The amount (as a percentage) to scale parent text size to determine warichu size. |

ParagraphStyle

Object available in ID 3, 4, 5, 6, 7

| PARAGRAPHSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|--|
| <code>allowArbitraryHyphenation</code> | <code>boolean allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. |

| PARAGRAPHSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|--|
| autoTcy | <code>const autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. |
| autoTcyIncludeRoman | <code>boolean autoTcyIncludeRoman</code> | If true, auto tcy includes Roman characters. |
| bunriKinshi | <code>boolean bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. |
| characterAlignment | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| characterRotation | <code>double characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. |
| cjkGridTracking | <code>boolean cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. |
| glyphForm | <code>#AlternateGlyphForms glyphForm</code> | The glyph variant to substitute for standard glyphs. |
| gridAlignment | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| gridGyoudori | <code>const gridGyoudori</code> | The manual gyoudori setting. |
| jidori | <code>const jidori</code> | The number of grid squares in which to arrange the text. |
| keepWithPrevious | <code>boolean keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. |
| kentenAlignment | <code>#KentenAlignment kentenAlignment</code> | The alignment of kenten characters relative to the parent characters. |
| kentenCharacterSet | <code>#KentenCharacterSet kentenCharacterSet</code> | The character set used for the custom kenten character. Note: Valid only when kenten kind is custom. |
| kentenCustomCharacter | <code>string kentenCustomCharacter</code> | The character used for kenten. Note: Valid only when kenten kind is custom. |
| kentenFillColor | <code>variant kentenFillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch or String. |
| kentenFont | <code>variant kentenFont</code> | The font to use for kenten characters. Type: Font or String. |
| kentenFontSize | <code>double kentenFontSize</code> | The size (in points) of kenten characters. |
| kentenFontStyle | <code>variant kentenFontStyle</code> | The font style of kenten characters. Type: String or idNothingEnum enumerator. |
| kentenKind | <code>#KentenCharacter kentenKind</code> | The style of kenten characters. |
| kentenOverprintFill | <code>#AdornmentOverprint kentenOverprintFill</code> | The method of overprinting the kenten fill. |
| kentenOverprintStroke | <code>#AdornmentOverprint kentenOverprintStroke</code> | The method of overprinting the kenten stroke. |
| kentenPlacement | <code>double kentenPlacement</code> | The distance between kenten characters and their parent characters. |
| kentenPosition | <code>#RubyKetenPosition kentenPosition</code> | The kenten position relative to the parent character. |
| kentenStrokeColor | <code>variant kentenStrokeColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of kenten characters. Type: Swatch or String. |
| kentenStrokeTint | <code>double kentenStrokeTint</code> | The stroke tint (as a percentage) of kenten characters. (Range: 0 to 100). |
| kentenTint | <code>double kentenTint</code> | The fill tint (as a percentage) of kenten characters. (Range: 0 to 100). |
| kentenWeight | <code>double kentenWeight</code> | The stroke weight (in points) of kenten characters. |
| kentenXScale | <code>double kentenXScale</code> | The horizontal size of kenten characters as a percent of the original size. |
| kentenYScale | <code>double kentenYScale</code> | The vertical size of kenten characters as a percent of the original size. |
| kinsokuHangType | <code>#KinsokuHangTypes kinsokuHangType</code> | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. |

| PARAGRAPHSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>kinsokuSet</code> | <code>variant kinsokuSet</code> | The <code>kinsoku</code> set that determines legitimate line breaks. Type: <code>KinsokuTable</code> , <code>idKinsokuSet</code> enumerator or <code>String</code> . |
| <code>kinsokuType</code> | <code>#KinsokuType kinsokuType</code> | The type of <code>kinsoku</code> processing for preventing <code>kinsoku</code> characters from beginning or ending a line. Note: Valid only when a <code>kinsoku</code> set is defined. |
| <code>leadingAki</code> | <code>double leadingAki</code> | The amount of space before each character. |
| <code>leadingModel</code> | <code>#LeadingModel leadingModel</code> | The point from which <code>leading</code> is measured from line to line. |
| <code>mojikumi</code> | <code>variant mojikumi</code> | The <code>mojikumi</code> table. For information, see <code>mojikumi</code> table defaults. Type: <code>MojikumiTable</code> , <code>String</code> or <code>idMojikumiTableDefaults</code> enumerator. |
| <code>otfHVKana</code> | <code>boolean otfHVKana</code> | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. |
| <code>otfProportionalMetrics</code> | <code>boolean otfProportionalMetrics</code> | If true, kerns according to proportional CJK metrics in OpenType fonts. |
| <code>otfRomanItalics</code> | <code>boolean otfRomanItalics</code> | If true, applies italics to half-width alphanumerics. |
| <code>paragraphGyoudori</code> | <code>boolean paragraphGyoudori</code> | If true, the <code>gyoudori</code> mode applies to the entire paragraph. If false, the <code>gyoudori</code> mode applies to each line in the paragraph. |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| <code>rensuuji</code> | <code>boolean rensuuji</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. |
| <code>rotateSingleByteCharacters</code> | <code>boolean rotateSingleByteCharacters</code> | If true, rotates Roman characters in vertical text. |
| <code>rubyAlignment</code> | <code>#RubyAlignments rubyAlignment</code> | The ruby alignment. |
| <code>rubyAutoAlign</code> | <code>boolean rubyAutoAlign</code> | If true, auto aligns ruby. |
| <code>rubyAutoScaling</code> | <code>boolean rubyAutoScaling</code> | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see <code>ruby</code> parent scaling percent. |
| <code>rubyAutoTcyAutoScale</code> | <code>boolean rubyAutoTcyAutoScale</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. |
| <code>rubyAutoTcyDigits</code> | <code>const rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. |
| <code>rubyAutoTcyIncludeRoman</code> | <code>boolean rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. |
| <code>rubyFill</code> | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: <code>Swatch</code> or <code>String</code> . |
| <code>rubyFont</code> | <code>variant rubyFont</code> | The font applied to ruby characters. Type: <code>Font</code> or <code>String</code> . |
| <code>rubyFontSize</code> | <code>double rubyFontSize</code> | The size (in points) of ruby characters. |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: <code>String</code> or <code>idNothingEnum</code> enumerator. |
| <code>rubyOpenTypePro</code> | <code>boolean rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. |
| <code>rubyOverhang</code> | <code>boolean rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see <code>ruby</code> parent overhang amount. |
| <code>rubyOverprintFill</code> | <code>#AdornmentOverprint rubyOverprintFill</code> | The method of overprinting the ruby fill. |
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | The method of overprinting the ruby stroke. |
| <code>rubyParentOverhangAmount</code> | <code>#RubyOverhang rubyParentOverhangAmount</code> | The amount by which ruby characters can overhang the parent text. |
| <code>rubyParentScalingPercent</code> | <code>double rubyParentScalingPercent</code> | The amount (as a percentage) to scale the parent text size to determine the ruby text size. |

| PARAGRAPHSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| <code>rubyParentSpacing</code> | <code>#RubyParentSpacing rubyParentSpacing</code> | <i>The ruby spacing relative to the parent text.</i> |
| <code>rubyPosition</code> | <code>#RubyKentePosition rubyPosition</code> | <i>The position of ruby characters relative to the parent text.</i> |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String.</i> |
| <code>rubyStrokeTint</code> | <code>double rubyStrokeTint</code> | <i>The stroke tint (as a percentage) of ruby characters.</i> |
| <code>rubyTint</code> | <code>double rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100).</i> |
| <code>rubyType</code> | <code>#RubyTypes rubyType</code> | <i>The ruby type.</i> |
| <code>rubyWeight</code> | <code>double rubyWeight</code> | <i>The stroke weight (in points) of ruby characters.</i> |
| <code>rubyXOffset</code> | <code>double rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters.</i> |
| <code>rubyXScale</code> | <code>double rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size.</i> |
| <code>rubyYOffset</code> | <code>double rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters.</i> |
| <code>rubyYScale</code> | <code>double rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size.</i> |
| <code>scaleAffectsLineHeight</code> | <code>boolean scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled.</i> |
| <code>shataiAdjustRotation</code> | <code>boolean shataiAdjustRotation</code> | <i>If true, applies shatai rotation.</i> |
| <code>shataiAdjustTsume</code> | <code>boolean shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume.</i> |
| <code>shataiDegreeAngle</code> | <code>double shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees).</i> |
| <code>shataiMagnification</code> | <code>double shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply.</i> |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | <i>The minimum space after a span or a split column. Type: Unit (Double or String).</i> |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | <i>The minimum space before a span or a split column. Type: Unit (Double or String).</i> |
| <code>spanColumnType</code> | <code>#SpanColumnTypeOptions spanColumnType</code> | <i>Whether a paragraph should be a single column, span columns or split columns.</i> |
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator.</i> |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | <i>The inside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>tatechuyoko</code> | <code>boolean tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text.</i> |
| <code>tatechuyokoXOffset</code> | <code>double tatechuyokoXOffset</code> | <i>The horizontal offset for horizontal characters in vertical text.</i> |
| <code>tatechuyokoYOffset</code> | <code>double tatechuyokoYOffset</code> | <i>The vertical offset for horizontal characters in vertical text.</i> |
| <code>trailingAki</code> | <code>double trailingAki</code> | <i>The amount of space after each character.</i> |
| <code>treatIdeographicSpaceAsSpace</code> | <code>boolean treatIdeographicSpaceAsSpace</code> | <i>If true, ideographic spaces will not wrap to the next line like text characters.</i> |
| <code>tsume</code> | <code>double tsume</code> | <i>The amount of horizontal character compression.</i> |
| <code>warichu</code> | <code>boolean warichu</code> | <i>If true, turns on warichu.</i> |
| <code>warichuAlignment</code> | <code>#WarichuAlignment warichuAlignment</code> | <i>The warichu alignment.</i> |

| PARAGRAPHSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|--|--|
| <code>warichuCharsAfterBreak</code> | <code>const warichuCharsAfterBreak</code> | <i>The minimum number of characters allowed after a line break.</i> |
| <code>warichuCharsBeforeBreak</code> | <code>const warichuCharsBeforeBreak</code> | <i>The minimum number of characters allowed before a line break.</i> |
| <code>warichuLines</code> | <code>const warichuLines</code> | <i>The number of lines of warichu within a single normal line.</i> |
| <code>warichuLineSpacing</code> | <code>double warichuLineSpacing</code> | <i>The gap between lines of warichu characters.</i> |
| <code>warichuSize</code> | <code>double warichuSize</code> | <i>The amount (as a percentage) to scale parent text size to determine warichu size.</i> |

PasteboardPreference

Object available in ID 3, 4, 5, 6, 7

| PASTEBOARDPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|--|
| <code>pasteboardMargins</code> | <code>variant pasteboardMargins</code> | <i>The minimum horizontal and vertical pasteboard margins. A horizontal margin of -1 means one document page width. Type: Array of 2 Units (Doubles or Strings).</i> |

PDF

Object available in ID 3, 4, 5, 6, 7

| PDF CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| <code>clearTransformations</code> | <code>void clearTransformations()</code> | <i>Clears transformations from the PDF. Transformations include rotation, scaling, flipping, fitting, and shearing.</i> |
| <code>convertShape</code> | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | <i>Converts the PDF to a different shape. / Given: The PDF's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0). Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points).</i> |
| <code>select</code> | <code>void select(#SelectionOptions ExistingSelection)</code> | <i>Selects the object. / ExistingSelection: The selection status of the PDF in relation to previously selected objects.</i> |
| <code>allGraphics</code> | <code>{Objects} *allGraphics</code> | <i>Lists all graphics contained by the PDF.</i> |
| <code>allPageItems</code> | <code>{Objects} *allPageItems</code> | <i>Lists all page items contained by the PDF.</i> |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | <i>The page item animation settings.</i> |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |

| PDF CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|---|--|
| bottomLeftCornerRadius | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| bottomRightCornerOption | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| bottomRightCornerRadius | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| endCap | <code>#EndCap endCap</code> | The end shape of an open path. |
| endJoin | <code>#EndJoin endJoin</code> | The corner join applied to the PDF. |
| fillColor | <code>variant fillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of the PDF. Type: Swatch or String. |
| fillTint | <code>double fillTint</code> | The percent of tint to use in the PDF's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| gapColor | <code>{Swatch} gapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type. |
| gapTint | <code>double gapTint</code> | The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| gradientFillAngle | <code>double gradientFillAngle</code> | The angle of a linear gradient applied to the fill of the PDF. (Range: -180 to 180). |
| gradientFillLength | <code>variant gradientFillLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the PDF. Type: Unit (Double or String). |
| gradientFillStart | <code>variant gradientFillStart</code> | The starting point (in page coordinates) of a gradient applied to the fill of the PDF, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| gradientStrokeAngle | <code>double gradientStrokeAngle</code> | The angle of a linear gradient applied to the stroke of the PDF. (Range: -180 to 180). |
| gradientStrokeLength | <code>variant gradientStrokeLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the PDF. Type: Unit (Double or String). |
| gradientStrokeStart | <code>variant gradientStrokeStart</code> | The starting point (in page coordinates) of a gradient applied to the stroke of the PDF, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| itemLayer | <code>{Layer} itemLayer</code> | The layer that the PDF is on. |
| leftLineEnd | <code>#ArrowHead leftLineEnd</code> | The arrowhead applied to the start of the path. |
| locked | <code>boolean locked</code> | If true, the PDF is locked. |
| miterLimit | <code>double miterLimit</code> | The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500). |
| name | <code>string name</code> | The name of the PDF. |
| overprintFill | <code>boolean overprintFill</code> | If true, the PDF's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors. |
| overprintGap | <code>boolean overprintGap</code> | If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors. |
| overprintStroke | <code>boolean overprintStroke</code> | If true, the PDF's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors. |
| parentPage | <code>{Page} *parentPage</code> | The page on which this page item appears. |
| preferences | <code>{Preferences} *preferences</code> | A collection of preferences objects. |

| PDF CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|---|--|
| <code>rightLineEnd</code> | <code>#ArrowHead rightLineEnd</code> | <i>The arrowhead applied to the end of the path.</i> |
| <code>strokeAlignment</code> | <code>#StrokeAlignment strokeAlignment</code> | <i>The stroke alignment applied to the PDF.</i> |
| <code>strokeColor</code> | <code>variant strokeColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the PDF. Type: Swatch or String.</i> |
| <code>strokeCornerAdjustment</code> | <code>#StrokeCornerAdjustment strokeCornerAdjustment</code> | <i>The corner adjustment applied to the PDF.</i> |
| <code>strokeDashAndGap</code> | <code>variant strokeDashAndGap</code> | <i>The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings).</i> |
| <code>strokeTint</code> | <code>double strokeTint</code> | <i>The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>strokeType</code> | <code>variant strokeType</code> | <i>The name of the stroke style to apply. Type: StrokeStyle or String.</i> |
| <code>strokeWeight</code> | <code>variant strokeWeight</code> | <i>The weight (in points) to apply to the PDF's stroke. Type: Unit (Double or String).</i> |
| <code>timingSettings</code> | <code>{TimingSetting} •timingSettings</code> | <i>The object timing settings.</i> |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the PDF is visible.</i> |

PDFExportPreference

Object available in ID 3, 4, 5, 6, 7

| PDFEXPORTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|--|
| <code>interactiveElementsOption</code> | <code>#InteractiveElementsOptions interactiveElementsOption</code> | <i>How to draw interactive elements.</i> |

PDFExportPreset

Object available in ID 3, 4, 5, 6, 7

| PDFEXPORTPRESET CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|--|
| <code>interactiveElementsOption</code> | <code>#InteractiveElementsOptions interactiveElementsOption</code> | <i>How to draw interactive elements.</i> |

PICT

Object available in ID 3, 4, 5, 6, 7

| PICT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|--|--|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| <code>clearTransformations</code> | <code>void clearTransformations()</code> | Clears transformations from the PICT. Transformations include rotation, scaling, flipping, fitting, and shearing. |
| <code>convertShape</code> | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | Converts the PICT to a different shape. / Given: The PICT's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0). Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points). |
| <code>select</code> | <code>void select(#SelectionOptions ExistingSelection)</code> | Selects the object. / ExistingSelection: The selection status of the PICT in relation to previously selected objects. |
| <code>allGraphics</code> | <code>{Objects} *allGraphics</code> | Lists all graphics contained by the PICT. |
| <code>allPageItems</code> | <code>{Objects} *allPageItems</code> | Lists all page items contained by the PICT. |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | The page item animation settings. |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | The shape to apply to the bottom left corner of rectangular shapes. |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| <code>endCap</code> | <code>#EndCap endCap</code> | The end shape of an open path. |
| <code>endJoin</code> | <code>#EndJoin endJoin</code> | The corner join applied to the PICT. |
| <code>fillColor</code> | <code>variant fillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of the PICT. . Type: Swatch or String. |
| <code>fillTint</code> | <code>double fillTint</code> | The percent of tint to use in the PICT's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| <code>gapColor</code> | <code>{Swatch} gapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type. |
| <code>gapTint</code> | <code>double gapTint</code> | The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| <code>gradientFillAngle</code> | <code>double gradientFillAngle</code> | The angle of a linear gradient applied to the fill of the PICT. (Range: -180 to 180). |

| PICT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|---|--|
| <code>gradientFillLength</code> | <code>variant gradientFillLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the PICT. Type: Unit (Double or String).</i> |
| <code>gradientFillStart</code> | <code>variant gradientFillStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the fill of the PICT, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>gradientStrokeAngle</code> | <code>double gradientStrokeAngle</code> | <i>The angle of a linear gradient applied to the stroke of the PICT. (Range: -180 to 180).</i> |
| <code>gradientStrokeLength</code> | <code>variant gradientStrokeLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the PICT. Type: Unit (Double or String).</i> |
| <code>gradientStrokeStart</code> | <code>variant gradientStrokeStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the stroke of the PICT, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>itemLayer</code> | <code>{Layer} itemLayer</code> | <i>The layer that the PICT is on.</i> |
| <code>leftLineEnd</code> | <code>#ArrowHead leftLineEnd</code> | <i>The arrowhead applied to the start of the path.</i> |
| <code>locked</code> | <code>boolean locked</code> | <i>If true, the PICT is locked.</i> |
| <code>miterLimit</code> | <code>double miterLimit</code> | <i>The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500).</i> |
| <code>name</code> | <code>string name</code> | <i>The name of the PICT.</i> |
| <code>overprintFill</code> | <code>boolean overprintFill</code> | <i>If true, the PICT's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors.</i> |
| <code>overprintGap</code> | <code>boolean overprintGap</code> | <i>If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors.</i> |
| <code>overprintStroke</code> | <code>boolean overprintStroke</code> | <i>If true, the PICT's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors.</i> |
| <code>parentPage</code> | <code>{Page} *parentPage</code> | <i>The page on which this page item appears.</i> |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| <code>rightLineEnd</code> | <code>#ArrowHead rightLineEnd</code> | <i>The arrowhead applied to the end of the path.</i> |
| <code>strokeAlignment</code> | <code>#StrokeAlignment strokeAlignment</code> | <i>The stroke alignment applied to the PICT.</i> |
| <code>strokeColor</code> | <code>variant strokeColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the PICT. Type: Swatch or String.</i> |
| <code>strokeCornerAdjustment</code> | <code>#StrokeCornerAdjustment strokeCornerAdjustment</code> | <i>The corner adjustment applied to the PICT.</i> |
| <code>strokeDashAndGap</code> | <code>variant strokeDashAndGap</code> | <i>The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings).</i> |
| <code>strokeTint</code> | <code>double strokeTint</code> | <i>The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>strokeType</code> | <code>variant strokeType</code> | <i>The name of the stroke style to apply. Type: StrokeStyle or String.</i> |
| <code>strokeWeight</code> | <code>variant strokeWeight</code> | <i>The weight (in points) to apply to the PICT's stroke. Type: Unit (Double or String).</i> |
| <code>timingSettings</code> | <code>{TimingSetting} *timingSettings</code> | <i>The object timing settings.</i> |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |

| PICT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|---|
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes.</i> Type: Unit (Double or String). |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes.</i> Type: Unit (Double or String). |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the PICT is visible.</i> |

Polygon

Object available in ID 3, 4, 5, 6, 7

| POLYGON CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file.</i> / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| <code>clearFrameFittingOptions</code> | <code>void clearFrameFittingOptions()</code> | <i>Removes the frame fittings options and resets it to the initial state.</i> |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | <i>The page item animation settings.</i> |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes.</i> Type: Unit (Double or String). |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes.</i> Type: Unit (Double or String). |
| <code>buttons</code> | <code>{Buttons} *buttons</code> | <i>A collection of buttons.</i> |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |
| <code>formFields</code> | <code>{FormFields} *formFields</code> | <i>A collection of form fields.</i> |
| <code>mediaItems</code> | <code>{MediaItems} *mediaItems</code> | <i>The media items collection.</i> |
| <code>multiStateObjects</code> | <code>{MultiStateObjects} *multiStateObjects</code> | <i>A collection of multi-state objects.</i> |
| <code>name</code> | <code>string name</code> | <i>The name of the Polygon.</i> |
| <code>parentPage</code> | <code>{Page} *parentPage</code> | <i>The page on which this page item appears.</i> |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |
| <code>timingSettings</code> | <code>{TimingSetting} *timingSettings</code> | <i>The object timing settings.</i> |

| POLYGON CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|--|
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the Polygon is visible.</i> |

RadiobuttonGroup

Object available in ID 3, 4, 5, 6, 7

| RADIOBUTTONGROUP CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|---------------------------------|---------------------------------|
| <code>widgets</code> | <code>{Widgets} *widgets</code> | <i>A collection of widgets.</i> |

Rectangle

Object available in ID 3, 4, 5, 6, 7

| RECTANGLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| <code>clearFrameFittingOptions</code> | <code>void clearFrameFittingOptions()</code> | <i>Removes the frame fittings options and resets it to the initial state.</i> |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | <i>The page item animation settings.</i> |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>buttons</code> | <code>{Buttons} *buttons</code> | <i>A collection of buttons.</i> |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |

| RECTANGLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|--|---|
| formFields | {FormFields} *formFields | A collection of form fields. |
| mediaItems | {MediaItems} *mediaItems | The media items collection. |
| multiStateObjects | {MultiStateObjects} *multiStateObjects | A collection of multi-state objects. |
| name | string name | The name of the Rectangle. |
| parentPage | {Page} *parentPage | The page on which this page item appears. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |
| splineItems | {SplineItems} *splineItems | The spline items collection. |
| timingSettings | {TimingSetting} *timingSettings | The object timing settings. |
| topLeftCornerOption | #CornerOptions topLeftCornerOption | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. |
| topLeftCornerRadius | variant topLeftCornerRadius | The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String). |
| topRightCornerOption | #CornerOptions topRightCornerOption | The shape to apply to the top right corner of rectangular shapes. |
| topRightCornerRadius | variant topRightCornerRadius | The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String). |
| visible | boolean visible | If true, the Rectangle is visible. |

Row

Object available in ID 3, 4, 5, 6, 7

| ROW CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|---|--|
| changeTransliterate | {Objects} changeTransliterate(variant ReverseOrder) | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| findTransliterate | {Objects} findTransliterate(variant ReverseOrder) | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| writingDirection | #HorizontalOrVertical writingDirection | The direction of the text in the cell. |

Sound

Object available in ID 3, 4, 5, 6, 7

| SOUND CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|---|
| applyObjectStyle | <code>void applyObjectStyle(ObjectStyle Using, boolean ClearingOverrides, boolean ClearingOverridesThroughRootObjectStyle)</code> | Applies the specified object style. / Using: The object style to apply. / ClearingOverrides: If true, clears the Sound's existing attributes before applying the style. / ClearingOverridesThroughRootObjectStyle: If true, clears attributes and formatting applied to the Sound that are not defined in the object style. |
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| changeObject | <code>{Objects} changeObject(variant ReverseOrder)</code> | Finds objects that match the find what value and replace the objects with the change to value. / Return value: The replaced object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| clearObjectStyleOverrides | <code>void clearObjectStyleOverrides()</code> | Clear overrides for object style. |
| export | <code>void export(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | Exports the object(s) to a file. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| findObject | <code>{Objects} findObject(variant ReverseOrder)</code> | Finds objects that match the find what value. / Return value: The found object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| flipItem | <code>void flipItem(#Flip Given, variant Around)</code> | Flips the Sound. / Given: The axis around which to flip the Sound. / Around: The point around which to flip the Sound. Type: Array of 2 Units (Doubles or Strings) or idAnchorPoint enumerator. |
| place | <code>variant place(string FileName, boolean ShowingOptions)</code> | Places the file. / Return value: The placed file. Type: Array of Pageitems, Graphics, Movies, Sounds, Stories or Texts. / FileName: The file to place. / ShowingOptions: Whether to display the import options dialog. |
| reframe | <code>void reframe(variant In, variant OpposingCorners)</code> | Move the bounding box of the page item. / In: The bounding box to resize. Type: idCoordinateSpaces enumerator or Ordered array containing CoordinateSpace:idCoordinateSpaces enumerator, BoundsKind:idBoundingBoxLimits enumerator. / OpposingCorners: Opposing corners of new bounding box in the given coordinate space. Type: Array of 2 Arrays of 2 Doubles. |
| resolve | <code>variant resolve(variant Location, #CoordinateSpaces In, boolean ConsideringRulerUnits)</code> | Get the coordinates of the given location in the specified coordinate system. / Return value: The resolved location for each of the targeted page items. Type: Array of 2 Doubles or Array of Arrays of 2 Doubles. / Location: The location requested. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / In: The coordinate space to use. / ConsideringRulerUnits: If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |

| SOUND CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>transformAgain</code> | <code>variant transformAgain()</code> | Transforms the Sound using the last transformation performed on any object. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| <code>transformAgainIndividually</code> | <code>variant transformAgainIndividually()</code> | Transforms the Sound using the last transformation performed on any Sound. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| <code>transformSequenceAgain</code> | <code>variant transformSequenceAgain()</code> | Transforms the Sound using the last sequence of transform operations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| <code>transformSequenceAgainIndividually</code> | <code>variant transformSequenceAgainIndividually()</code> | Transforms the Sound using the last sequence of transformations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| <code>transformValuesOf</code> | <code>{Objects} transformValuesOf(#CoordinateSpaces In)</code> | Get the transformation values of the page item. / Return value: A list of the transformation matrices. / In: The coordinate space to use. |
| <code>absoluteFlip</code> | <code>#Flip absoluteFlip</code> | Indicates whether the Sound has been flipped independently of its parent object and, if yes, the direction in which the Sound was flipped. |
| <code>absoluteHorizontalScale</code> | <code>double absoluteHorizontalScale</code> | The horizontal scale of the Sound relative to its containing object. |
| <code>absoluteRotationAngle</code> | <code>double absoluteRotationAngle</code> | The rotation angle of the Sound relative to its containing object. (Range: -360 to 360). |
| <code>absoluteShearAngle</code> | <code>double absoluteShearAngle</code> | The skewing angle of the Sound relative to its containing object. (Range: -360 to 360). |
| <code>absoluteVerticalScale</code> | <code>double absoluteVerticalScale</code> | The vertical scale of the Sound relative to its containing object. |
| <code>allGraphics</code> | <code>{Objects} *allGraphics</code> | Lists all graphics contained by the Sound. |
| <code>allPageItems</code> | <code>{Objects} *allPageItems</code> | Lists all page items contained by the Sound. |
| <code>animationSettings</code> | <code>{AnimationSetting} *animationSettings</code> | The page item animation settings. |
| <code>appliedObjectStyle</code> | <code>{ObjectStyle} appliedObjectStyle</code> | The object style applied to the Sound. |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | The shape to apply to the bottom left corner of rectangular shapes. |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| <code>contentTransparencySettings</code> | <code>{ContentTransparencySetting} *contentTransparencySettings</code> | Transparency settings for the content of the Sound. |
| <code>endCap</code> | <code>#EndCap endCap</code> | The end shape of an open path. |
| <code>endJoin</code> | <code>#EndJoin endJoin</code> | The corner join applied to the Sound. |
| <code>fillColor</code> | <code>variant fillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of the Sound. . Type: Swatch or String. |
| <code>fillTint</code> | <code>double fillTint</code> | The percent of tint to use in the Sound's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| <code>fillTransparencySettings</code> | <code>{FillTransparencySetting} *fillTransparencySettings</code> | Transparency settings for the fill applied to the Sound. |

| SOUND CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|---|--|
| <code>flip</code> | <code>#Flip flip</code> | <i>The direction in which to flip the printed image.</i> |
| <code>gapColor</code> | <code>{Swatch} gapColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type.</i> |
| <code>gapTint</code> | <code>double gapTint</code> | <i>The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>gradientFillAngle</code> | <code>double gradientFillAngle</code> | <i>The angle of a linear gradient applied to the fill of the Sound. (Range: -180 to 180).</i> |
| <code>gradientFillLength</code> | <code>variant gradientFillLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the Sound. Type: Unit (Double or String).</i> |
| <code>gradientFillStart</code> | <code>variant gradientFillStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the fill of the Sound, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>gradientStrokeAngle</code> | <code>double gradientStrokeAngle</code> | <i>The angle of a linear gradient applied to the stroke of the Sound. (Range: -180 to 180).</i> |
| <code>gradientStrokeLength</code> | <code>variant gradientStrokeLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the Sound. Type: Unit (Double or String).</i> |
| <code>gradientStrokeStart</code> | <code>variant gradientStrokeStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the stroke of the Sound, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>graphics</code> | <code>{Graphics} •graphics</code> | <i>A collection of imported graphics in any graphic file format (vector, metafile, or bitmap).</i> |
| <code>leftLineEnd</code> | <code>#ArrowHead leftLineEnd</code> | <i>The arrowhead applied to the start of the path.</i> |
| <code>localDisplaySetting</code> | <code>#DisplaySettingOptions localDisplaySetting</code> | <i>Display performance options for the Sound.</i> |
| <code>miterLimit</code> | <code>double miterLimit</code> | <i>The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500).</i> |
| <code>nonprinting</code> | <code>boolean nonprinting</code> | <i>If true, the Sound does not print.</i> |
| <code>overprintFill</code> | <code>boolean overprintFill</code> | <i>If true, the Sound's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors.</i> |
| <code>overprintGap</code> | <code>boolean overprintGap</code> | <i>If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors.</i> |
| <code>overprintStroke</code> | <code>boolean overprintStroke</code> | <i>If true, the Sound's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors.</i> |
| <code>pageItems</code> | <code>{PageItems} •pageItems</code> | <i>The page items collection, which can be used to process all page items in a container (such as a document, page, or group), regardless of type.</i> |
| <code>parentPage</code> | <code>{Page} •parentPage</code> | <i>The page on which this page item appears.</i> |
| <code>paths</code> | <code>{Paths} •paths</code> | <i>A collection of paths.</i> |
| <code>preferences</code> | <code>{Preferences} •preferences</code> | <i>A collection of preferences objects.</i> |
| <code>rightLineEnd</code> | <code>#ArrowHead rightLineEnd</code> | <i>The arrowhead applied to the end of the path.</i> |
| <code>rotationAngle</code> | <code>double rotationAngle</code> | <i>The rotation angle of the Sound. (Range: -360 to 360).</i> |
| <code>shearAngle</code> | <code>double shearAngle</code> | <i>The skewing angle applied to the Sound. (Range: -360 to 360).</i> |
| <code>soundLoop</code> | <code>boolean soundLoop</code> | <i>If true, sound loops forever (SWF only).</i> |
| <code>stopOnPageTurn</code> | <code>boolean stopOnPageTurn</code> | <i>If true, sounds stops playing when its page turns.</i> |

| SOUND CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|--|
| strokeAlignment | <code>#StrokeAlignment</code> strokeAlignment | <i>The stroke alignment applied to the Sound.</i> |
| strokeColor | <code>variant</code> strokeColor | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the Sound. Type: Swatch or String.</i> |
| strokeCornerAdjustment | <code>#StrokeCornerAdjustment</code> strokeCornerAdjustment | <i>The corner adjustment applied to the Sound.</i> |
| strokeDashAndGap | <code>variant</code> strokeDashAndGap | <i>The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings).</i> |
| strokeTint | <code>double</code> strokeTint | <i>The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| strokeTransparencySettings | <code>{StrokeTransparencySetting}</code> *strokeTransparencySettings | <i>Transparency settings for the stroke.</i> |
| strokeType | <code>variant</code> strokeType | <i>The name of the stroke style to apply. Type: StrokeStyle or String.</i> |
| strokeWeight | <code>variant</code> strokeWeight | <i>The weight (in points) to apply to the Sound's stroke. Type: Unit (Double or String).</i> |
| textWrapPreferences | <code>{TextWrapPreference}</code> *textWrapPreferences | <i>The text wrap preference properties that define the default formatting for wrapping text around objects.</i> |
| timingSettings | <code>{TimingSetting}</code> *timingSettings | <i>The object timing settings.</i> |
| topLeftCornerOption | <code>#CornerOptions</code> topLeftCornerOption | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| topLeftCornerRadius | <code>variant</code> topLeftCornerRadius | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| topRightCornerOption | <code>#CornerOptions</code> topRightCornerOption | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| topRightCornerRadius | <code>variant</code> topRightCornerRadius | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| transparencySettings | <code>{TransparencySetting}</code> *transparencySettings | <i>Transparency settings.</i> |
| visible | <code>boolean</code> visible | <i>If true, the Sound is visible.</i> |

Spread

Object available in ID 3, 4, 5, 6, 7

| SPREAD CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|---|--|
| select | <code>void</code> select(<code>#SelectionOptions</code> ExistingSelection) | <i>Selects the object. / ExistingSelection: The selection status of the Spread in relation to previously selected objects.</i> |
| epsTexts | <code>{EPSTexts}</code> *epsTexts | <i>EPSTexts.</i> |
| multiStateObjects | <code>{MultiStateObjects}</code> *multiStateObjects | <i>A collection of multi-state objects.</i> |
| name | <code>string</code> name | <i>The name of the Spread; this is an alias to the Spread's label property.</i> |
| preferences | <code>{Preferences}</code> *preferences | <i>A collection of preferences objects.</i> |

| SPREAD CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------|---------------------------------|------------------------------|
| splineItems | {SplineItems} *splineItems | The spline items collection. |
| timingSettings | {TimingSetting} *timingSettings | The object timing settings. |

State

Object available in ID 3, 4, 5, 6, 7

| STATE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------|---|---|
| addItemsToState | void addItemsToState(Objects PageItems) | Adds page items to this state. / Pageitems: One or more page items to add to this state. |
| move | void move(integer NewPosition) | Moves the state to a new position in its parent collection. / NewPosition: the index to move the state to in its parent collection. |
| releaseAsObject | void releaseAsObject() | Releases this state's appearance as a page item, removing the state from its parent object. |
| epsTexts | {EPSTexts} *epsTexts | EPSTexts. |
| splineItems | {SplineItems} *splineItems | The spline items collection. |

Story

Object available in ID 3, 4, 5, 6, 7

| STORY CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|---|--|
| asynchronousExportFile | {BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave) | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| changeTransliterate | {Objects} changeTransliterate(variant ReverseOrder) | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| findTransliterate | {Objects} findTransliterate(variant ReverseOrder) | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| allowArbitraryHyphenation | boolean allowArbitraryHyphenation | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. |
| appliedNamedGrid | {NamedGrid} appliedNamedGrid | The named grid in use. |
| autoTcy | const autoTcy | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. |
| autoTcyIncludeRoman | boolean autoTcyIncludeRoman | If true, auto tcy includes Roman characters. |

| STORY CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------|--|--|
| bunriKinshi | <code>boolean bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. |
| characterAlignment | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| characterRotation | <code>double characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. |
| cjkGridTracking | <code>boolean cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. |
| epsTexts | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| glyphForm | <code>#AlternateGlyphForms glyphForm</code> | The glyph variant to substitute for standard glyphs. |
| gridAlignment | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| gridData | <code>{GridDataInformation} *gridData</code> | Default grid properties. Note: Applies to named, layout, and frame (story) grids. |
| gridGyoudori | <code>const gridGyoudori</code> | The manual gyoudori setting. |
| jidori | <code>const jidori</code> | The number of grid squares in which to arrange the text. |
| keepWithPrevious | <code>boolen keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. |
| kentenAlignment | <code>#KentenAlignment kentenAlignment</code> | The alignment of kenten characters relative to the parent characters. |
| kentenCharacterSet | <code>#KentenCharacterSet kentenCharacterSet</code> | The character set used for the custom kenten character. Note: Valid only when kenten kind is custom. |
| kentenCustomCharacter | <code>string kentenCustomCharacter</code> | The character used for kenten. Note: Valid only when kenten kind is custom. |
| kentenFillColor | <code>variant kentenFillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch or String. |
| kentenFont | <code>variant kentenFont</code> | The font to use for kenten characters. Type: Font or String. |
| kentenFontSize | <code>double kentenFontSize</code> | The size (in points) of kenten characters. |
| kentenFontStyle | <code>variant kentenFontStyle</code> | The font style of kenten characters. Type: String or idNothingEnum enumerator. |
| kentenKind | <code>#KentenCharacter kentenKind</code> | The style of kenten characters. |
| kentenOverprintFill | <code>#AdornmentOverprint kentenOverprintFill</code> | The method of overprinting the kenten fill. |
| kentenOverprintStroke | <code>#AdornmentOverprint kentenOverprintStroke</code> | The method of overprinting the kenten stroke. |
| kentenPlacement | <code>double kentenPlacement</code> | The distance between kenten characters and their parent characters. |
| kentenPosition | <code>#RubyKetenPosition kentenPosition</code> | The kenten position relative to the parent character. |
| kentenStrokeColor | <code>variant kentenStrokeColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of kenten characters. Type: Swatch or String. |
| kentenStrokeTint | <code>double kentenStrokeTint</code> | The stroke tint (as a percentage) of kenten characters. (Range: 0 to 100). |
| kentenTint | <code>double kentenTint</code> | The fill tint (as a percentage) of kenten characters. (Range: 0 to 100). |
| kentenWeight | <code>double kentenWeight</code> | The stroke weight (in points) of kenten characters. |
| kentenXScale | <code>double kentenXScale</code> | The horizontal size of kenten characters as a percent of the original size. |
| kentenYScale | <code>double kentenYScale</code> | The vertical size of kenten characters as a percent of the original size. |
| kinsokuHangType | <code>#KinsokuHangTypes kinsokuHangType</code> | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. |

| STORY CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>kinsokuSet</code> | <code>variant kinsokuSet</code> | The <code>kinsoku</code> set that determines legitimate line breaks. Type: <code>KinsokuTable</code> , <code>idKinsokuSet</code> enumerator or <code>String</code> . |
| <code>kinsokuType</code> | <code>#KinsokuType kinsokuType</code> | The type of <code>kinsoku</code> processing for preventing <code>kinsoku</code> characters from beginning or ending a line. Note: Valid only when a <code>kinsoku</code> set is defined. |
| <code>leadingAki</code> | <code>double leadingAki</code> | The amount of space before each character. |
| <code>leadingModel</code> | <code>#LeadingModel leadingModel</code> | The point from which <code>leading</code> is measured from line to line. |
| <code>mojikumi</code> | <code>variant mojikumi</code> | The <code>mojikumi</code> table. For information, see <code>mojikumi</code> table defaults. Type: <code>MojikumiTable</code> , <code>String</code> or <code>idMojikumiTableDefaults</code> enumerator. |
| <code>multiStateObjects</code> | <code>{MultiStateObjects} *multiStateObjects</code> | A collection of multi-state objects. |
| <code>name</code> | <code>string name</code> | The name of the Story; this is an alias to the Story's label property. |
| <code>otfHVKana</code> | <code>boolean otfHVKana</code> | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. |
| <code>otfProportionalMetrics</code> | <code>boolean otfProportionalMetrics</code> | If true, kerns according to proportional CJK metrics in OpenType fonts. |
| <code>otfRomanItalics</code> | <code>boolean otfRomanItalics</code> | If true, applies italics to half-width alphanumerics. |
| <code>paragraphGyoudori</code> | <code>boolean paragraphGyoudori</code> | If true, the <code>gyoudori</code> mode applies to the entire paragraph. If false, the <code>gyoudori</code> mode applies to each line in the paragraph. |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| <code>rensuuji</code> | <code>boolean rensuuji</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. |
| <code>rotateSingleByteCharacters</code> | <code>boolean rotateSingleByteCharacters</code> | If true, rotates Roman characters in vertical text. |
| <code>rubyAlignment</code> | <code>#RubyAlignments rubyAlignment</code> | The ruby alignment. |
| <code>rubyAutoAlign</code> | <code>boolean rubyAutoAlign</code> | If true, auto aligns ruby. |
| <code>rubyAutoScaling</code> | <code>boolean rubyAutoScaling</code> | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see <code>ruby</code> parent scaling percent. |
| <code>rubyAutoTcyAutoSize</code> | <code>boolean rubyAutoTcyAutoSize</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. |
| <code>rubyAutoTcyDigits</code> | <code>const rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. |
| <code>rubyAutoTcyIncludeRoman</code> | <code>boolean rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. |
| <code>rubyFill</code> | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: <code>Swatch</code> or <code>String</code> . |
| <code>rubyFlag</code> | <code>boolean rubyFlag</code> | If true, ruby is on. |
| <code>rubyFont</code> | <code>variant rubyFont</code> | The font applied to ruby characters. Type: <code>Font</code> or <code>String</code> . |
| <code>rubyFontSize</code> | <code>double rubyFontSize</code> | The size (in points) of ruby characters. |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: <code>String</code> or <code>idNothingEnum</code> enumerator. |
| <code>rubyOpenTypePro</code> | <code>boolean rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. |
| <code>rubyOverhang</code> | <code>boolean rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see <code>ruby</code> parent overhang amount. |
| <code>rubyOverprintFill</code> | <code>#AdornmentOverprint rubyOverprintFill</code> | The method of overprinting the ruby fill. |
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | The method of overprinting the ruby stroke. |

| STORY CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---|--|
| rubyParentOverhangAmount | <code>#RubyOverhang rubyParentOverhangAmount</code> | The amount by which ruby characters can overhang the parent text. |
| rubyParentScalingPercent | <code>double rubyParentScalingPercent</code> | The amount (as a percentage) to scale the parent text size to determine the ruby text size. |
| rubyParentSpacing | <code>#RubyParentSpacing rubyParentSpacing</code> | The ruby spacing relative to the parent text. |
| rubyPosition | <code>#RubyKetenPosition rubyPosition</code> | The position of ruby characters relative to the parent text. |
| rubyString | <code>string rubyString</code> | The ruby string contents. |
| rubyStroke | <code>variant rubyStroke</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String. |
| rubyStrokeTint | <code>double rubyStrokeTint</code> | The stroke tint (as a percentage) of ruby characters. |
| rubyTint | <code>double rubyTint</code> | The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). |
| rubyType | <code>#RubyTypes rubyType</code> | The ruby type. |
| rubyWeight | <code>double rubyWeight</code> | The stroke weight (in points) of ruby characters. |
| rubyXOffset | <code>double rubyXOffset</code> | The amount of horizontal space between ruby and parent characters. |
| rubyXScale | <code>double rubyXScale</code> | The horizontal size of ruby characters, specified as a percent of the original size. |
| rubyYOffset | <code>double rubyYOffset</code> | The amount of vertical space between ruby and parent characters. |
| rubyYScale | <code>double rubyYScale</code> | The vertical size of ruby characters, specified as a percent of the original size. |
| scaleAffectsLineHeight | <code>boolean scaleAffectsLineHeight</code> | If true, the line changes size when characters are scaled. |
| shataiAdjustRotation | <code>boolean shataiAdjustRotation</code> | If true, applies shatai rotation. |
| shataiAdjustTsume | <code>boolean shataiAdjustTsume</code> | If true, adjusts shatai tsume. |
| shataiDegreeAngle | <code>double shataiDegreeAngle</code> | The shatai lens angle (in degrees). |
| shataiMagnification | <code>double shataiMagnification</code> | The amount (as a percentage) of shatai obliquing to apply. |
| spanColumnMinSpaceAfter | <code>variant spanColumnMinSpaceAfter</code> | The minimum space after a span or a split column. Type: Unit (Double or String). |
| spanColumnMinSpaceBefore | <code>variant spanColumnMinSpaceBefore</code> | The minimum space before a span or a split column. Type: Unit (Double or String). |
| spanColumnType | <code>#SpanColumnTypeOptions spanColumnType</code> | Whether a paragraph should be a single column, span columns or split columns. |
| spanSplitColumnCount | <code>variant spanSplitColumnCount</code> | The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator. |
| splineItems | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| splitColumnInsideGutter | <code>variant splitColumnInsideGutter</code> | The inside gutter if the paragraph splits columns. Type: Unit (Double or String). |
| splitColumnOutsideGutter | <code>variant splitColumnOutsideGutter</code> | The outside gutter if the paragraph splits columns. Type: Unit (Double or String). |
| tatechuyoko | <code>boolean tatechuyoko</code> | If true, makes the character horizontal in vertical text. |
| tatechuyokoXOffset | <code>double tatechuyokoXOffset</code> | The horizontal offset for horizontal characters in vertical text. |
| tatechuyokoYOffset | <code>double tatechuyokoYOffset</code> | The vertical offset for horizontal characters in vertical text. |

| STORY CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| <code>trailingAki</code> | <code>double trailingAki</code> | <i>The amount of space after each character.</i> |
| <code>treatIdeographicSpaceAsSpace</code> | <code>boolean treatIdeographicSpaceAsSpace</code> | <i>If true, ideographic spaces will not wrap to the next line like text characters.</i> |
| <code>tsume</code> | <code>double tsume</code> | <i>The amount of horizontal character compression.</i> |
| <code>warichu</code> | <code>boolean warichu</code> | <i>If true, turns on warichu.</i> |
| <code>warichuAlignment</code> | <code>#WarichuAlignment warichuAlignment</code> | <i>The warichu alignment.</i> |
| <code>warichuCharsAfterBreak</code> | <code>const warichuCharsAfterBreak</code> | <i>The minimum number of characters allowed after a line break.</i> |
| <code>warichuCharsBeforeBreak</code> | <code>const warichuCharsBeforeBreak</code> | <i>The minimum number of characters allowed before a line break.</i> |
| <code>warichuLines</code> | <code>const warichuLines</code> | <i>The number of lines of warichu within a single normal line.</i> |
| <code>warichuLineSpacing</code> | <code>double warichuLineSpacing</code> | <i>The gap between lines of warichu characters.</i> |
| <code>warichuSize</code> | <code>double warichuSize</code> | <i>The amount (as a percentage) to scale parent text size to determine warichu size.</i> |

Table

Object available in ID 3, 4, 5, 6, 7

| TABLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|---|
| <code>changeTransliterate</code> | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | <i>Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean.</i> |
| <code>findTransliterate</code> | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | <i>Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean.</i> |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |
| <code>multiStateObjects</code> | <code>{MultiStateObjects} *multiStateObjects</code> | <i>A collection of multi-state objects.</i> |
| <code>name</code> | <code>string name</code> | <i>The name of the Table; this is an alias to the Table's label property.</i> |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |

Text

Object available in ID 3, 4, 5, 6, 7

| TEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |

| TEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|--|
| <code>changeTransliterate</code> | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the <code>find</code> character type value and replaces the text with the <code>change</code> character type value. / Return value: The replaced text. / <code>ReverseOrder</code> : If true, returns the results in reverse order. Type: Boolean. |
| <code>findHyperlinks</code> | <code>{Objects} findHyperlinks(variant SortOrder)</code> | Finds hyperlink sources that intersecting with specified text range. / Return value: The found hyperlinks. / <code>SortOrder</code> : The sort order of found ranges. Type: <code>idRangeSortOrder</code> enumerator. |
| <code>findTransliterate</code> | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the <code>find</code> character type value. / Return value: The found text. / <code>ReverseOrder</code> : If true, returns the results in reverse order. Type: Boolean. |
| <code>showText</code> | <code>void showText()</code> | Jump to the text range. |
| <code>allowArbitraryHyphenation</code> | <code>boolean allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. |
| <code>autoTcy</code> | <code>const autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. |
| <code>autoTcyIncludeRoman</code> | <code>boolean autoTcyIncludeRoman</code> | If true, <code>autoTcy</code> includes Roman characters. |
| <code>bunriKinshi</code> | <code>boolean bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. |
| <code>characterAlignment</code> | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| <code>characterRotation</code> | <code>double characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. |
| <code>cjkGridTracking</code> | <code>boolean cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| <code>glyphForm</code> | <code>#AlternateGlyphForms glyphForm</code> | The <code>glyph</code> variant to substitute for standard glyphs. |
| <code>gridAlignment</code> | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| <code>gridGyoudori</code> | <code>const gridGyoudori</code> | The manual gyoudori setting. |
| <code>jidori</code> | <code>const jidori</code> | The number of grid squares in which to arrange the text. |
| <code>keepWithPrevious</code> | <code>boolean keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. |
| <code>kentenAlignment</code> | <code>#KntenAlignment kentenAlignment</code> | The alignment of kenten characters relative to the parent characters. |
| <code>kentenCharacterSet</code> | <code>#KntenCharacterSet kentenCharacterSet</code> | The character set used for the custom kenten character. Note: Valid only when <code>kenten kind</code> is custom. |
| <code>kentenCustomCharacter</code> | <code>string kentenCustomCharacter</code> | The character used for kenten. Note: Valid only when <code>kenten kind</code> is custom. |
| <code>kentenFillColor</code> | <code>variant kentenFillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch or String. |
| <code>kentenFont</code> | <code>variant kentenFont</code> | The font to use for kenten characters. Type: Font or String. |
| <code>kentenFontSize</code> | <code>double kentenFontSize</code> | The size (in points) of kenten characters. |
| <code>kentenFontStyle</code> | <code>variant kentenFontStyle</code> | The font style of kenten characters. Type: String or <code>idNothingEnum</code> enumerator. |
| <code>kentenKind</code> | <code>#KntenCharacter kentenKind</code> | The style of kenten characters. |
| <code>kentenOverprintFill</code> | <code>#AdornmentOverprint kentenOverprintFill</code> | The method of overprinting the kenten fill. |
| <code>kentenOverprintStroke</code> | <code>#AdornmentOverprint kentenOverprintStroke</code> | The method of overprinting the kenten stroke. |

| TEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|---|---|
| kntenPlacement | <code>double</code> kntenPlacement | <i>The distance between knten characters and their parent characters.</i> |
| kntenPosition | <code>#RubyKntenPosition</code> kntenPosition | <i>The knten position relative to the parent character.</i> |
| kntenStrokeColor | <code>variant</code> kntenStrokeColor | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of knten characters. Type: Swatch or String.</i> |
| kntenStrokeTint | <code>double</code> kntenStrokeTint | <i>The stroke tint (as a percentage) of knten characters. (Range: 0 to 100).</i> |
| kntenTint | <code>double</code> kntenTint | <i>The fill tint (as a percentage) of knten characters. (Range: 0 to 100).</i> |
| kntenWeight | <code>double</code> kntenWeight | <i>The stroke weight (in points) of knten characters.</i> |
| kntenXScale | <code>double</code> kntenXScale | <i>The horizontal size of knten characters as a percent of the original size.</i> |
| kntenYScale | <code>double</code> kntenYScale | <i>The vertical size of knten characters as a percent of the original size.</i> |
| kinsokuHangType | <code>#KinsokuHangTypes</code> kinsokuHangType | <i>The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect.</i> |
| kinsokuSet | <code>variant</code> kinsokuSet | <i>The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator or String.</i> |
| kinsokuType | <code>#KinsokuType</code> kinsokuType | <i>The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined.</i> |
| leadingAki | <code>double</code> leadingAki | <i>The amount of space before each character.</i> |
| leadingModel | <code>#LeadingModel</code> leadingModel | <i>The point from which leading is measured from line to line.</i> |
| mojikumi | <code>variant</code> mojikumi | <i>The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String or idMojikumiTableDefaults enumerator.</i> |
| multiStateObjects | <code>{MultiStateObjects}</code> *multiStateObjects | <i>A collection of multi-state objects.</i> |
| otfHVKana | <code>boolean</code> otfHVKana | <i>If true, switches hiragana fonts, which have different glyphs for horizontal and vertical.</i> |
| otfProportionalMetrics | <code>boolean</code> otfProportionalMetrics | <i>If true, kerns according to proportional CJK metrics in OpenType fonts.</i> |
| otfRomanItalics | <code>boolean</code> otfRomanItalics | <i>If true, applies italics to half-width alphanumerics.</i> |
| paragraphGyoudori | <code>boolean</code> paragraphGyoudori | <i>If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph.</i> |
| preferences | <code>{Preferences}</code> *preferences | <i>A collection of preferences objects.</i> |
| rensuuji | <code>boolean</code> renssuuji | <i>If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers.</i> |
| rotateSingleByteCharacters | <code>boolean</code> rotateSingleByteCharacters | <i>If true, rotates Roman characters in vertical text.</i> |
| rubyAlignment | <code>#RubyAlignments</code> rubyAlignment | <i>The ruby alignment.</i> |
| rubyAutoAlign | <code>boolean</code> rubyAutoAlign | <i>If true, auto aligns ruby.</i> |
| rubyAutoScaling | <code>boolean</code> rubyAutoScaling | <i>If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent.</i> |
| rubyAutoTcyAutoSize | <code>boolean</code> rubyAutoTcyAutoSize | <i>If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em.</i> |
| rubyAutoTcyDigits | <code>const</code> rubyAutoTcyDigits | <i>The number of digits included in auto tcy (tate-chuu-yoko) in ruby.</i> |
| rubyAutoTcyIncludeRoman | <code>boolean</code> rubyAutoTcyIncludeRoman | <i>If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby.</i> |
| rubyFill | <code>variant</code> rubyFill | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch or String.</i> |

| TEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|--|--|
| <code>rubyFlag</code> | <code>boolean rubyFlag</code> | If true, ruby is on. |
| <code>rubyFont</code> | <code>variant rubyFont</code> | The font applied to ruby characters. Type: Font or String. |
| <code>rubyFontSize</code> | <code>double rubyFontSize</code> | The size (in points) of ruby characters. |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: String or idNothingEnum enumerator. |
| <code>rubyOpenTypePro</code> | <code>boolean rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. |
| <code>rubyOverhang</code> | <code>boolean rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. |
| <code>rubyOverprintFill</code> | <code>#AdornmentOverprint rubyOverprintFill</code> | The method of overprinting the ruby fill. |
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | The method of overprinting the ruby stroke. |
| <code>rubyParentOverhangAmount</code> | <code>#RubyOverhang rubyParentOverhangAmount</code> | The amount by which ruby characters can overhang the parent text. |
| <code>rubyParentScalingPercent</code> | <code>double rubyParentScalingPercent</code> | The amount (as a percentage) to scale the parent text size to determine the ruby text size. |
| <code>rubyParentSpacing</code> | <code>#RubyParentSpacing rubyParentSpacing</code> | The ruby spacing relative to the parent text. |
| <code>rubyPosition</code> | <code>#RubyKetenPosition rubyPosition</code> | The position of ruby characters relative to the parent text. |
| <code>rubyString</code> | <code>string rubyString</code> | The ruby string contents. |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String. |
| <code>rubyStrokeTint</code> | <code>double rubyStrokeTint</code> | The stroke tint (as a percentage) of ruby characters. |
| <code>rubyTint</code> | <code>double rubyTint</code> | The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). |
| <code>rubyType</code> | <code>#RubyTypes rubyType</code> | The ruby type. |
| <code>rubyWeight</code> | <code>double rubyWeight</code> | The stroke weight (in points) of ruby characters. |
| <code>rubyXOffset</code> | <code>double rubyXOffset</code> | The amount of horizontal space between ruby and parent characters. |
| <code>rubyXScale</code> | <code>double rubyXScale</code> | The horizontal size of ruby characters, specified as a percent of the original size. |
| <code>rubyYOffset</code> | <code>double rubyYOffset</code> | The amount of vertical space between ruby and parent characters. |
| <code>rubyYScale</code> | <code>double rubyYScale</code> | The vertical size of ruby characters, specified as a percent of the original size. |
| <code>scaleAffectsLineHeight</code> | <code>boolean scaleAffectsLineHeight</code> | If true, the line changes size when characters are scaled. |
| <code>shataiAdjustRotation</code> | <code>boolean shataiAdjustRotation</code> | If true, applies shatai rotation. |
| <code>shataiAdjustTsume</code> | <code>boolean shataiAdjustTsume</code> | If true, adjusts shatai tsume. |
| <code>shataiDegreeAngle</code> | <code>double shataiDegreeAngle</code> | The shatai lens angle (in degrees). |
| <code>shataiMagnification</code> | <code>double shataiMagnification</code> | The amount (as a percentage) of shatai obliquing to apply. |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | The minimum space after a span or a split column. Type: Unit (Double or String). |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | The minimum space before a span or a split column. Type: Unit (Double or String). |
| <code>spanColumnType</code> | <code>#SpanColumnTypeOptions spanColumnType</code> | Whether a paragraph should be a single column, span columns or split columns. |

| TEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|---|
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns.</i> Type: Integer (1 - 40) or <code>idSpanColumnCountOptions</code> enumerator. |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | <i>The inside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>tatechuyoko</code> | <code>boolean tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text.</i> |
| <code>tatechuyokoXOffset</code> | <code>double tatechuyokoXOffset</code> | <i>The horizontal offset for horizontal characters in vertical text.</i> |
| <code>tatechuyokoYOffset</code> | <code>double tatechuyokoYOffset</code> | <i>The vertical offset for horizontal characters in vertical text.</i> |
| <code>trailingAki</code> | <code>double trailingAki</code> | <i>The amount of space after each character.</i> |
| <code>treatIdeographicSpaceAsSpace</code> | <code>boolean treatIdeographicSpaceAsSpace</code> | <i>If true, ideographic spaces will not wrap to the next line like text characters.</i> |
| <code>tsume</code> | <code>double tsume</code> | <i>The amount of horizontal character compression.</i> |
| <code>warichu</code> | <code>boolean warichu</code> | <i>If true, turns on warichu.</i> |
| <code>warichuAlignment</code> | <code>#WarichuAlignment warichuAlignment</code> | <i>The warichu alignment.</i> |
| <code>warichuCharsAfterBreak</code> | <code>const warichuCharsAfterBreak</code> | <i>The minimum number of characters allowed after a line break.</i> |
| <code>warichuCharsBeforeBreak</code> | <code>const warichuCharsBeforeBreak</code> | <i>The minimum number of characters allowed before a line break.</i> |
| <code>warichuLines</code> | <code>const warichuLines</code> | <i>The number of lines of warichu within a single normal line.</i> |
| <code>warichuLineSpacing</code> | <code>double warichuLineSpacing</code> | <i>The gap between lines of warichu characters.</i> |
| <code>warichuSize</code> | <code>double warichuSize</code> | <i>The amount (as a percentage) to scale parent text size to determine warichu size.</i> |

TextColumn

Object available in ID 3, 4, 5, 6, 7

| TEXTCOLUMN CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: <code>idExportFormat</code> enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: <code>PDFExportPreset</code>. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| <code>changeTransliterate</code> | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | <i>Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean.</i> |
| <code>findHyperlinks</code> | <code>{Objects} findHyperlinks(variant SortOrder)</code> | <i>Finds hyperlink sources that intersecting with specified text range. / Return value: The found hyperlinks. / SortOrder: The sort order of found ranges. Type: <code>idRangeSortOrder</code> enumerator.</i> |

| TextColumn CS5 Members | Prototype | Description |
|--|--|---|
| <code>findTransliterate</code> | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the <code>find</code> character type value. / Return value: The found text. / <code>ReverseOrder</code> : If true, returns the results in reverse order. Type: Boolean. |
| <code>showText</code> | <code>void showText()</code> | Jump to the text range. |
| <code>allowArbitraryHyphenation</code> | <code>boolean allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. |
| <code>autoTcy</code> | <code>const autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. |
| <code>autoTcyIncludeRoman</code> | <code>boolean autoTcyIncludeRoman</code> | If true, <code>autoTcy</code> includes Roman characters. |
| <code>bunriKinshi</code> | <code>boolean bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (-) to the selected <code>kinsoku</code> set. Note: Valid only when a <code>kinsoku</code> set is in effect. |
| <code>characterAlignment</code> | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| <code>characterRotation</code> | <code>double characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. |
| <code>cjkGridTracking</code> | <code>boolean cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| <code>glyphForm</code> | <code>#AlternateGlyphForms glyphForm</code> | The <code>glyph</code> variant to substitute for standard glyphs. |
| <code>gridAlignment</code> | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| <code>gridGyoudori</code> | <code>const gridGyoudori</code> | The manual <code>gyoudori</code> setting. |
| <code>jidori</code> | <code>const jidori</code> | The number of grid squares in which to arrange the text. |
| <code>keepWithPrevious</code> | <code>boolean keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. |
| <code>kentenAlignment</code> | <code>#KentenAlignment kentenAlignment</code> | The alignment of <code>kenten</code> characters relative to the parent characters. |
| <code>kentenCharacterSet</code> | <code>#KentenCharacterSet kentenCharacterSet</code> | The character set used for the custom <code>kenten</code> character. Note: Valid only when <code>kenten kind</code> is <code>custom</code> . |
| <code>kentenCustomCharacter</code> | <code>string kentenCustomCharacter</code> | The character used for <code>kenten</code> . Note: Valid only when <code>kenten kind</code> is <code>custom</code> . |
| <code>kentenFillColor</code> | <code>variant kentenFillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of <code>kenten</code> characters. Type: Swatch or String. |
| <code>kentenFont</code> | <code>variant kentenFont</code> | The font to use for <code>kenten</code> characters. Type: Font or String. |
| <code>kentenFontSize</code> | <code>double kentenFontSize</code> | The size (in points) of <code>kenten</code> characters. |
| <code>kentenFontStyle</code> | <code>variant kentenFontStyle</code> | The font style of <code>kenten</code> characters. Type: String or <code>idNothingEnum</code> enumerator. |
| <code>kentenKind</code> | <code>#KentenCharacter kentenKind</code> | The style of <code>kenten</code> characters. |
| <code>kentenOverprintFill</code> | <code>#AdornmentOverprint kentenOverprintFill</code> | The method of overprinting the <code>kenten</code> fill. |
| <code>kentenOverprintStroke</code> | <code>#AdornmentOverprint kentenOverprintStroke</code> | The method of overprinting the <code>kenten</code> stroke. |
| <code>kentenPlacement</code> | <code>double kentenPlacement</code> | The distance between <code>kenten</code> characters and their parent characters. |
| <code>kentenPosition</code> | <code>#RubyKentenPosition kentenPosition</code> | The <code>kenten</code> position relative to the parent character. |
| <code>kentenStrokeColor</code> | <code>variant kentenStrokeColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of <code>kenten</code> characters. Type: Swatch or String. |
| <code>kentenStrokeTint</code> | <code>double kentenStrokeTint</code> | The stroke tint (as a percentage) of <code>kenten</code> characters. (Range: 0 to 100). |

| TextColumn CS5 Members | Prototype | Description |
|----------------------------|--|--|
| kentenTint | double kentenTint | The fill tint (as a percentage) of kenten characters. (Range: 0 to 100). |
| kentenWeight | double kentenWeight | The stroke weight (in points) of kenten characters. |
| kentenXScale | double kentenXScale | The horizontal size of kenten characters as a percent of the original size. |
| kentenYScale | double kentenYScale | The vertical size of kenten characters as a percent of the original size. |
| kinsokuHangType | #KinsokuHangTypes kinsokuHangType | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. |
| kinsokuSet | variant kinsokuSet | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator or String. |
| kinsokuType | #KinsokuType kinsokuType | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. |
| leadingAki | double leadingAki | The amount of space before each character. |
| leadingModel | #LeadingModel leadingModel | The point from which leading is measured from line to line. |
| mojikumi | variant mojikumi | The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String or idMojikumiTableDefaults enumerator. |
| multiStateObjects | {MultiStateObjects} *multiStateObjects | A collection of multi-state objects. |
| otfHVKana | boolean otfHVKana | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. |
| otfProportionalMetrics | boolean otfProportionalMetrics | If true, kerns according to proportional CJK metrics in OpenType fonts. |
| otfRomanItalics | boolean otfRomanItalics | If true, applies italics to half-width alphanumerics. |
| paragraphGyoudori | boolean paragraphGyoudori | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |
| rensuuji | boolean rensuuji | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. |
| rotateSingleByteCharacters | boolean rotateSingleByteCharacters | If true, rotates Roman characters in vertical text. |
| rubyAlignment | #RubyAlignments rubyAlignment | The ruby alignment. |
| rubyAutoAlign | boolean rubyAutoAlign | If true, auto aligns ruby. |
| rubyAutoScaling | boolean rubyAutoScaling | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. |
| rubyAutoTcyAutoScale | boolean rubyAutoTcyAutoScale | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. |
| rubyAutoTcyDigits | const rubyAutoTcyDigits | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. |
| rubyAutoTcyIncludeRoman | boolean rubyAutoTcyIncludeRoman | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. |
| rubyFill | variant rubyFill | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch or String. |
| rubyFlag | boolean rubyFlag | If true, ruby is on. |
| rubyFont | variant rubyFont | The font applied to ruby characters. Type: Font or String. |
| rubyFontSize | double rubyFontSize | The size (in points) of ruby characters. |
| rubyFontStyle | variant rubyFontStyle | The font style of ruby characters. Type: String or idNothingEnum enumerator. |

| TextColumn CS5 Members | Prototype | Description |
|---------------------------------------|--|--|
| <code>rubyOpenTypePro</code> | <code>boolean rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. |
| <code>rubyOverhang</code> | <code>boolean rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. |
| <code>rubyOverprintFill</code> | <code>#AdornmentOverprint rubyOverprintFill</code> | The method of overprinting the ruby fill. |
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | The method of overprinting the ruby stroke. |
| <code>rubyParentOverhangAmount</code> | <code>#RubyOverhang rubyParentOverhangAmount</code> | The amount by which ruby characters can overhang the parent text. |
| <code>rubyParentScalingPercent</code> | <code>double rubyParentScalingPercent</code> | The amount (as a percentage) to scale the parent text size to determine the ruby text size. |
| <code>rubyParentSpacing</code> | <code>#RubyParentSpacing rubyParentSpacing</code> | The ruby spacing relative to the parent text. |
| <code>rubyPosition</code> | <code>#RubyKertenPosition rubyPosition</code> | The position of ruby characters relative to the parent text. |
| <code>rubyString</code> | <code>string rubyString</code> | The ruby string contents. |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String. |
| <code>rubyStrokeTint</code> | <code>double rubyStrokeTint</code> | The stroke tint (as a percentage) of ruby characters. |
| <code>rubyTint</code> | <code>double rubyTint</code> | The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). |
| <code>rubyType</code> | <code>#RubyTypes rubyType</code> | The ruby type. |
| <code>rubyWeight</code> | <code>double rubyWeight</code> | The stroke weight (in points) of ruby characters. |
| <code>rubyXOffset</code> | <code>double rubyXOffset</code> | The amount of horizontal space between ruby and parent characters. |
| <code>rubyXScale</code> | <code>double rubyXScale</code> | The horizontal size of ruby characters, specified as a percent of the original size. |
| <code>rubyYOffset</code> | <code>double rubyYOffset</code> | The amount of vertical space between ruby and parent characters. |
| <code>rubyYScale</code> | <code>double rubyYScale</code> | The vertical size of ruby characters, specified as a percent of the original size. |
| <code>scaleAffectsLineHeight</code> | <code>boolean scaleAffectsLineHeight</code> | If true, the line changes size when characters are scaled. |
| <code>shataiAdjustRotation</code> | <code>boolean shataiAdjustRotation</code> | If true, applies shatai rotation. |
| <code>shataiAdjustTsume</code> | <code>boolean shataiAdjustTsume</code> | If true, adjusts shatai tsume. |
| <code>shataiDegreeAngle</code> | <code>double shataiDegreeAngle</code> | The shatai lens angle (in degrees). |
| <code>shataiMagnification</code> | <code>double shataiMagnification</code> | The amount (as a percentage) of shatai obliquing to apply. |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | The minimum space after a span or a split column. Type: Unit (Double or String). |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | The minimum space before a span or a split column. Type: Unit (Double or String). |
| <code>spanColumnType</code> | <code>#SpanColumnTypeOptions spanColumnType</code> | Whether a paragraph should be a single column, span columns or split columns. |
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator. |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | The inside gutter if the paragraph splits columns. Type: Unit (Double or String). |

| TEXTCOLUMN CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|---|
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>tatechuyoko</code> | <code>boolean tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text.</i> |
| <code>tatechuyokoXOffset</code> | <code>double tatechuyokoXOffset</code> | <i>The horizontal offset for horizontal characters in vertical text.</i> |
| <code>tatechuyokoYOffset</code> | <code>double tatechuyokoYOffset</code> | <i>The vertical offset for horizontal characters in vertical text.</i> |
| <code>trailingAki</code> | <code>double trailingAki</code> | <i>The amount of space after each character.</i> |
| <code>treatIdeographicSpaceAsSpace</code> | <code>boolean treatIdeographicSpaceAsSpace</code> | <i>If true, ideographic spaces will not wrap to the next line like text characters.</i> |
| <code>tsume</code> | <code>double tsume</code> | <i>The amount of horizontal character compression.</i> |
| <code>warichu</code> | <code>boolean warichu</code> | <i>If true, turns on warichu.</i> |
| <code>warichuAlignment</code> | <code>#WarichuAlignment warichuAlignment</code> | <i>The warichu alignment.</i> |
| <code>warichuCharsAfterBreak</code> | <code>const warichuCharsAfterBreak</code> | <i>The minimum number of characters allowed after a line break.</i> |
| <code>warichuCharsBeforeBreak</code> | <code>const warichuCharsBeforeBreak</code> | <i>The minimum number of characters allowed before a line break.</i> |
| <code>warichuLines</code> | <code>const warichuLines</code> | <i>The number of lines of warichu within a single normal line.</i> |
| <code>warichuLineSpacing</code> | <code>double warichuLineSpacing</code> | <i>The gap between lines of warichu characters.</i> |
| <code>warichuSize</code> | <code>double warichuSize</code> | <i>The amount (as a percentage) to scale parent text size to determine warichu size.</i> |

TextDefault

Object available in ID 3, 4, 5, 6, 7

| TEXTDEFAULT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|---|---|
| <code>allowArbitraryHyphenation</code> | <code>boolean allowArbitraryHyphenation</code> | <i>If true, words unassociated with a hyphenation dictionary can break to the next line on any character.</i> |
| <code>appliedNamedGrid</code> | <code>{NamedGrid} appliedNamedGrid</code> | <i>The named grid in use.</i> |
| <code>autoTcy</code> | <code>const autoTcy</code> | <i>The number of half-width characters at or below which the characters automatically run horizontally in vertical text.</i> |
| <code>autoTcyIncludeRoman</code> | <code>boolean autoTcyIncludeRoman</code> | <i>If true, auto tcy includes Roman characters.</i> |
| <code>bunriKinshi</code> | <code>boolean bunriKinshi</code> | <i>If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect.</i> |
| <code>characterAlignment</code> | <code>#CharacterAlignment characterAlignment</code> | <i>The alignment of small characters to the largest character in the line.</i> |
| <code>characterRotation</code> | <code>double characterRotation</code> | <i>The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise.</i> |
| <code>cjkGridTracking</code> | <code>boolean cjkGridTracking</code> | <i>If true, uses grid tracking to track non-Roman characters in CJK grids.</i> |
| <code>glyphForm</code> | <code>#AlternateGlyphForms glyphForm</code> | <i>The glyph variant to substitute for standard glyphs.</i> |
| <code>gridAlignment</code> | <code>#GridAlignment gridAlignment</code> | <i>The alignment to the frame grid or baseline grid.</i> |
| <code>gridGyoudori</code> | <code>const gridGyoudori</code> | <i>The manual gyoudori setting.</i> |
| <code>jidori</code> | <code>const jidori</code> | <i>The number of grid squares in which to arrange the text.</i> |

| TEXTDEFAULT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|---|---|
| <code>keepWithPrevious</code> | <code>boolean keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. |
| <code>kntenAlignment</code> | <code>#KntenAlignment kntenAlignment</code> | The alignment of knten characters relative to the parent characters. |
| <code>kntenCharacterSet</code> | <code>#KntenCharacterSet kntenCharacterSet</code> | The character set used for the custom knten character. Note: Valid only when knten kind is custom. |
| <code>kntenCustomCharacter</code> | <code>string kntenCustomCharacter</code> | The character used for knten. Note: Valid only when knten kind is custom. |
| <code>kntenFillColor</code> | <code>variant kntenFillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of knten characters. Type: Swatch or String. |
| <code>kntenFont</code> | <code>variant kntenFont</code> | The font to use for knten characters. Type: Font or String. |
| <code>kntenFontSize</code> | <code>double kntenFontSize</code> | The size (in points) of knten characters. |
| <code>kntenFontStyle</code> | <code>variant kntenFontStyle</code> | The font style of knten characters. Type: String or idNothingEnum enumerator. |
| <code>kntenKind</code> | <code>#KntenCharacter kntenKind</code> | The style of knten characters. |
| <code>kntenOverprintFill</code> | <code>#AdornmentOverprint kntenOverprintFill</code> | The method of overprinting the knten fill. |
| <code>kntenOverprintStroke</code> | <code>#AdornmentOverprint kntenOverprintStroke</code> | The method of overprinting the knten stroke. |
| <code>kntenPlacement</code> | <code>double kntenPlacement</code> | The distance between knten characters and their parent characters. |
| <code>kntenPosition</code> | <code>#RubyKetenPosition kntenPosition</code> | The knten position relative to the parent character. |
| <code>kntenStrokeColor</code> | <code>variant kntenStrokeColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of knten characters. Type: Swatch or String. |
| <code>kntenStrokeTint</code> | <code>double kntenStrokeTint</code> | The stroke tint (as a percentage) of knten characters. (Range: 0 to 100). |
| <code>kntenTint</code> | <code>double kntenTint</code> | The fill tint (as a percentage) of knten characters. (Range: 0 to 100). |
| <code>kntenWeight</code> | <code>double kntenWeight</code> | The stroke weight (in points) of knten characters. |
| <code>kntenXScale</code> | <code>double kntenXScale</code> | The horizontal size of knten characters as a percent of the original size. |
| <code>kntenYScale</code> | <code>double kntenYScale</code> | The vertical size of knten characters as a percent of the original size. |
| <code>kinsokuHangType</code> | <code>#KinsokuHangTypes kinsokuHangType</code> | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. |
| <code>kinsokuSet</code> | <code>variant kinsokuSet</code> | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator or String. |
| <code>kinsokuType</code> | <code>#KinsokuType kinsokuType</code> | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. |
| <code>leadingAki</code> | <code>double leadingAki</code> | The amount of space before each character. |
| <code>leadingModel</code> | <code>#LeadingModel leadingModel</code> | The point from which leading is measured from line to line. |
| <code>mojikumi</code> | <code>variant mojikumi</code> | The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String or idMojikumiTableDefaults enumerator. |
| <code>otfHVKana</code> | <code>boolean otfHVKana</code> | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. |
| <code>otfProportionalMetrics</code> | <code>boolean otfProportionalMetrics</code> | If true, kerns according to proportional CJK metrics in OpenType fonts. |
| <code>otfRomanItalics</code> | <code>boolean otfRomanItalics</code> | If true, applies italics to half-width alphanumerics. |
| <code>paragraphGyoudori</code> | <code>boolean paragraphGyoudori</code> | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. |

| TEXTDEFAULT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>preferences</code> | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| <code>rensuumi</code> | <code>boolean rensuumi</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. |
| <code>rotateSingleByteCharacters</code> | <code>boolean rotateSingleByteCharacters</code> | If true, rotates Roman characters in vertical text. |
| <code>rubyAlignment</code> | <code>#RubyAlignments rubyAlignment</code> | The ruby alignment. |
| <code>rubyAutoAlign</code> | <code>boolean rubyAutoAlign</code> | If true, auto aligns ruby. |
| <code>rubyAutoScaling</code> | <code>boolean rubyAutoScaling</code> | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. |
| <code>rubyAutoTcyAutoScale</code> | <code>boolean rubyAutoTcyAutoScale</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. |
| <code>rubyAutoTcyDigits</code> | <code>const rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. |
| <code>rubyAutoTcyIncludeRoman</code> | <code>boolean rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. |
| <code>rubyFill</code> | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch or String. |
| <code>rubyFont</code> | <code>variant rubyFont</code> | The font applied to ruby characters. Type: Font or String. |
| <code>rubyFontSize</code> | <code>double rubyFontSize</code> | The size (in points) of ruby characters. |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: String or idNothingEnum enumerator. |
| <code>rubyOpenTypePro</code> | <code>boolean rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. |
| <code>rubyOverhang</code> | <code>boolean rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. |
| <code>rubyOverprintFill</code> | <code>#AdornmentOverprint rubyOverprintFill</code> | The method of overprinting the ruby fill. |
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | The method of overprinting the ruby stroke. |
| <code>rubyParentOverhangAmount</code> | <code>#RubyOverhang rubyParentOverhangAmount</code> | The amount by which ruby characters can overhang the parent text. |
| <code>rubyParentScalingPercent</code> | <code>double rubyParentScalingPercent</code> | The amount (as a percentage) to scale the parent text size to determine the ruby text size. |
| <code>rubyParentSpacing</code> | <code>#RubyParentSpacing rubyParentSpacing</code> | The ruby spacing relative to the parent text. |
| <code>rubyPosition</code> | <code>#RubyKetenPosition rubyPosition</code> | The position of ruby characters relative to the parent text. |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String. |
| <code>rubyStrokeTint</code> | <code>double rubyStrokeTint</code> | The stroke tint (as a percentage) of ruby characters. |
| <code>rubyTint</code> | <code>double rubyTint</code> | The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). |
| <code>rubyType</code> | <code>#RubyTypes rubyType</code> | The ruby type. |
| <code>rubyWeight</code> | <code>double rubyWeight</code> | The stroke weight (in points) of ruby characters. |
| <code>rubyXOffset</code> | <code>double rubyXOffset</code> | The amount of horizontal space between ruby and parent characters. |
| <code>rubyXScale</code> | <code>double rubyXScale</code> | The horizontal size of ruby characters, specified as a percent of the original size. |
| <code>rubyYOffset</code> | <code>double rubyYOffset</code> | The amount of vertical space between ruby and parent characters. |

| TEXTDEFAULT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| <code>rubyYScale</code> | <code>double rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size.</i> |
| <code>scaleAffectsLineHeight</code> | <code>boolean scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled.</i> |
| <code>shataiAdjustRotation</code> | <code>boolean shataiAdjustRotation</code> | <i>If true, applies shatai rotation.</i> |
| <code>shataiAdjustTsume</code> | <code>boolean shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume.</i> |
| <code>shataiDegreeAngle</code> | <code>double shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees).</i> |
| <code>shataiMagnification</code> | <code>double shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply.</i> |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | <i>The minimum space after a span or a split column. Type: Unit (Double or String).</i> |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | <i>The minimum space before a span or a split column. Type: Unit (Double or String).</i> |
| <code>spanColumnType</code> | <code>#SpanColumnTypeOptions spanColumnType</code> | <i>Whether a paragraph should be a single column, span columns or split columns.</i> |
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator.</i> |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | <i>The inside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>tatechuyoko</code> | <code>boolean tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text.</i> |
| <code>tatechuyokoXOffset</code> | <code>double tatechuyokoXOffset</code> | <i>The horizontal offset for horizontal characters in vertical text.</i> |
| <code>tatechuyokoYOffset</code> | <code>double tatechuyokoYOffset</code> | <i>The vertical offset for horizontal characters in vertical text.</i> |
| <code>trailingAki</code> | <code>double trailingAki</code> | <i>The amount of space after each character.</i> |
| <code>treatIdeographicSpaceAsSpace</code> | <code>boolean treatIdeographicSpaceAsSpace</code> | <i>If true, ideographic spaces will not wrap to the next line like text characters.</i> |
| <code>tsume</code> | <code>double tsume</code> | <i>The amount of horizontal character compression.</i> |
| <code>warichu</code> | <code>boolean warichu</code> | <i>If true, turns on warichu.</i> |
| <code>warichuAlignment</code> | <code>#WarichuAlignment warichuAlignment</code> | <i>The warichu alignment.</i> |
| <code>warichuCharsAfterBreak</code> | <code>const warichuCharsAfterBreak</code> | <i>The minimum number of characters allowed after a line break.</i> |
| <code>warichuCharsBeforeBreak</code> | <code>const warichuCharsBeforeBreak</code> | <i>The minimum number of characters allowed before a line break.</i> |
| <code>warichuLines</code> | <code>const warichuLines</code> | <i>The number of lines of warichu within a single normal line.</i> |
| <code>warichuLineSpacing</code> | <code>double warichuLineSpacing</code> | <i>The gap between lines of warichu characters.</i> |
| <code>warichuSize</code> | <code>double warichuSize</code> | <i>The amount (as a percentage) to scale parent text size to determine warichu size.</i> |

TextFrame

Object available in ID 3, 4, 5, 6, 7

| TEXTFRAME CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------|---|--|
| asynchronousExportFile | {BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave) | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| changeTransliterate | {Objects} changeTransliterate(variant ReverseOrder) | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| findTransliterate | {Objects} findTransliterate(variant ReverseOrder) | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| animationSettings | {AnimationSetting} *animationSettings | The page item animation settings. |
| bottomLeftCornerOption | #CornerOptions bottomLeftCornerOption | The shape to apply to the bottom left corner of rectangular shapes. |
| bottomLeftCornerRadius | variant bottomLeftCornerRadius | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| bottomRightCornerOption | #CornerOptions bottomRightCornerOption | The shape to apply to the bottom right corner of rectangular shapes. |
| bottomRightCornerRadius | variant bottomRightCornerRadius | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| epsTexts | {EPSTexts} *epsTexts | EPSTexts. |
| gridData | {GridDataInformation} *gridData | Default grid properties. Note: Applies to named, layout, and frame (story) grids. |
| multiStateObjects | {MultiStateObjects} *multiStateObjects | A collection of multi-state objects. |
| name | string name | The name of the TextFrame. |
| parentPage | {Page} *parentPage | The page on which this page item appears. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |
| splineItems | {SplineItems} *splineItems | The spline items collection. |
| timingSettings | {TimingSetting} *timingSettings | The object timing settings. |
| topLeftCornerOption | #CornerOptions topLeftCornerOption | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. |
| topLeftCornerRadius | variant topLeftCornerRadius | The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String). |
| topRightCornerOption | #CornerOptions topRightCornerOption | The shape to apply to the top right corner of rectangular shapes. |
| topRightCornerRadius | variant topRightCornerRadius | The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String). |
| visible | boolean visible | If true, the TextFrame is visible. |

TextFramePreference

Object available in ID 3, 4, 5, 6, 7

| TEXTFRAMEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---|---|
| verticalBalanceColumns | <code>boolean verticalBalanceColumns</code> | Vertically justify balanced across all columns. |

TextPath

Object available in ID 3, 4, 5, 6, 7

| TEXTPATH CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------|--|--|
| changeTransliterate | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| findTransliterate | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| name | <code>string name</code> | The name of the TextPath; this is an alias to the TextPath's label property. |

TextPreference

Object available in ID 3, 4, 5, 6, 7

| TEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|---|
| highlightKinsoku | <code>boolean highlightKinsoku</code> | If true, uses on-screen highlighting to identify kinsoku. |
| useCidMojikumi | <code>boolean useCidMojikumi</code> | If true, uses the glyph CID to get the mojikumi class of the character. |
| useNewVerticalScaling | <code>boolean useNewVerticalScaling</code> | If true, reverses X and Y scaling on Roman characters in vertical text. |

TextStyleRange

Object available in ID 3, 4, 5, 6, 7

| TEXTSTYLERANGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|--|
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| changeTransliterate | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |

| TEXTSTYLERANGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|--|
| findHyperlinks | <code>{Objects} findHyperlinks(variant SortOrder)</code> | Finds hyperlink sources that intersecting with specified text range. / Return value: The found hyperlinks. / SortOrder: The sort order of found ranges. Type: idRangeSortOrder enumerator. |
| findTransliterate | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| showText | <code>void showText()</code> | Jump to the text range. |
| allowArbitraryHyphenation | <code>boolean allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. |
| autoTcy | <code>const autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. |
| autoTcyIncludeRoman | <code>boolean autoTcyIncludeRoman</code> | If true, auto tcy includes Roman characters. |
| bunriKinshi | <code>boolean bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. |
| characterAlignment | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| characterRotation | <code>double characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. |
| cjkGridTracking | <code>boolean cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. |
| epsTexts | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| glyphForm | <code>#AlternateGlyphForms glyphForm</code> | The glyph variant to substitute for standard glyphs. |
| gridAlignment | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| gridGyoudori | <code>const gridGyoudori</code> | The manual gyoudori setting. |
| jidori | <code>const jidori</code> | The number of grid squares in which to arrange the text. |
| keepWithPrevious | <code>boolean keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. |
| kentenAlignment | <code>#KentenAlignment kentenAlignment</code> | The alignment of kenten characters relative to the parent characters. |
| kentenCharacterSet | <code>#KentenCharacterSet kentenCharacterSet</code> | The character set used for the custom kenten character. Note: Valid only when kenten kind is custom. |
| kentenCustomCharacter | <code>string kentenCustomCharacter</code> | The character used for kenten. Note: Valid only when kenten kind is custom. |
| kentenFillColor | <code>variant kentenFillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch or String. |
| kentenFont | <code>variant kentenFont</code> | The font to use for kenten characters. Type: Font or String. |
| kentenFontSize | <code>double kentenFontSize</code> | The size (in points) of kenten characters. |
| kentenFontStyle | <code>variant kentenFontStyle</code> | The font style of kenten characters. Type: String or idNothingEnum enumerator. |
| kentenKind | <code>#KentenCharacter kentenKind</code> | The style of kenten characters. |
| kentenOverprintFill | <code>#AdornmentOverprint kentenOverprintFill</code> | The method of overprinting the kenten fill. |
| kentenOverprintStroke | <code>#AdornmentOverprint kentenOverprintStroke</code> | The method of overprinting the kenten stroke. |
| kentenPlacement | <code>double kentenPlacement</code> | The distance between kenten characters and their parent characters. |
| kentenPosition | <code>#RubyKentenPosition kentenPosition</code> | The kenten position relative to the parent character. |

| TEXTSTYLERANGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|---|--|
| kntenStrokeColor | <code>variant</code> kntenStrokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of knten characters. Type: Swatch or String. |
| kntenStrokeTint | <code>double</code> kntenStrokeTint | The stroke tint (as a percentage) of knten characters. (Range: 0 to 100). |
| kntenTint | <code>double</code> kntenTint | The fill tint (as a percentage) of knten characters. (Range: 0 to 100). |
| kntenWeight | <code>double</code> kntenWeight | The stroke weight (in points) of knten characters. |
| kntenXScale | <code>double</code> kntenXScale | The horizontal size of knten characters as a percent of the original size. |
| kntenYScale | <code>double</code> kntenYScale | The vertical size of knten characters as a percent of the original size. |
| kinsokuHangType | <code>#KinsokuHangTypes</code> kinsokuHangType | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. |
| kinsokuSet | <code>variant</code> kinsokuSet | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator or String. |
| kinsokuType | <code>#KinsokuType</code> kinsokuType | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. |
| leadingAki | <code>double</code> leadingAki | The amount of space before each character. |
| leadingModel | <code>#LeadingModel</code> leadingModel | The point from which leading is measured from line to line. |
| mojikumi | <code>variant</code> mojikumi | The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String or idMojikumiTableDefaults enumerator. |
| multiStateObjects | <code>{MultiStateObjects}</code> •multiStateObjects | A collection of multi-state objects. |
| otfHVKana | <code>boolean</code> otfHVKana | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. |
| otfProportionalMetrics | <code>boolean</code> otfProportionalMetrics | If true, kerns according to proportional CJK metrics in OpenType fonts. |
| otfRomanItalics | <code>boolean</code> otfRomanItalics | If true, applies italics to half-width alphanumerics. |
| paragraphGyoudori | <code>boolean</code> paragraphGyoudori | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. |
| preferences | <code>{Preferences}</code> •preferences | A collection of preferences objects. |
| rensuuji | <code>boolean</code> rensuuji | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. |
| rotateSingleByteCharacters | <code>boolean</code> rotateSingleByteCharacters | If true, rotates Roman characters in vertical text. |
| rubyAlignment | <code>#RubyAlignments</code> rubyAlignment | The ruby alignment. |
| rubyAutoAlign | <code>boolean</code> rubyAutoAlign | If true, auto aligns ruby. |
| rubyAutoScaling | <code>boolean</code> rubyAutoScaling | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. |
| rubyAutoTcyAutoSize | <code>boolean</code> rubyAutoTcyAutoSize | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. |
| rubyAutoTcyDigits | <code>const</code> rubyAutoTcyDigits | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. |
| rubyAutoTcyIncludeRoman | <code>boolean</code> rubyAutoTcyIncludeRoman | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. |
| rubyFill | <code>variant</code> rubyFill | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch or String. |
| rubyFlag | <code>boolean</code> rubyFlag | If true, ruby is on. |
| rubyFont | <code>variant</code> rubyFont | The font applied to ruby characters. Type: Font or String. |

| TEXTSTYLERANGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|--|---|
| rubyFontSize | <code>double rubyFontSize</code> | <i>The size (in points) of ruby characters.</i> |
| rubyFontStyle | <code>variant rubyFontStyle</code> | <i>The font style of ruby characters. Type: String or idNothingEnum enumerator.</i> |
| rubyOpenTypePro | <code>boolean rubyOpenTypePro</code> | <i>If true, uses OpenType Pro fonts for ruby.</i> |
| rubyOverhang | <code>boolean rubyOverhang</code> | <i>If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount.</i> |
| rubyOverprintFill | <code>#AdornmentOverprint rubyOverprintFill</code> | <i>The method of overprinting the ruby fill.</i> |
| rubyOverprintStroke | <code>#AdornmentOverprint rubyOverprintStroke</code> | <i>The method of overprinting the ruby stroke.</i> |
| rubyParentOverhangAmount | <code>#RubyOverhang rubyParentOverhangAmount</code> | <i>The amount by which ruby characters can overhang the parent text.</i> |
| rubyParentScalingPercent | <code>double rubyParentScalingPercent</code> | <i>The amount (as a percentage) to scale the parent text size to determine the ruby text size.</i> |
| rubyParentSpacing | <code>#RubyParentSpacing rubyParentSpacing</code> | <i>The ruby spacing relative to the parent text.</i> |
| rubyPosition | <code>#RubyKentePosition rubyPosition</code> | <i>The position of ruby characters relative to the parent text.</i> |
| rubyString | <code>string rubyString</code> | <i>The ruby string contents.</i> |
| rubyStroke | <code>variant rubyStroke</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String.</i> |
| rubyStrokeTint | <code>double rubyStrokeTint</code> | <i>The stroke tint (as a percentage) of ruby characters.</i> |
| rubyTint | <code>double rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100).</i> |
| rubyType | <code>#RubyTypes rubyType</code> | <i>The ruby type.</i> |
| rubyWeight | <code>double rubyWeight</code> | <i>The stroke weight (in points) of ruby characters.</i> |
| rubyXOffset | <code>double rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters.</i> |
| rubyXScale | <code>double rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size.</i> |
| rubyYOffset | <code>double rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters.</i> |
| rubyYScale | <code>double rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size.</i> |
| scaleAffectsLineHeight | <code>boolean scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled.</i> |
| shataiAdjustRotation | <code>boolean shataiAdjustRotation</code> | <i>If true, applies shatai rotation.</i> |
| shataiAdjustTsume | <code>boolean shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume.</i> |
| shataiDegreeAngle | <code>double shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees).</i> |
| shataiMagnification | <code>double shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply.</i> |
| spanColumnMinSpaceAfter | <code>variant spanColumnMinSpaceAfter</code> | <i>The minimum space after a span or a split column. Type: Unit (Double or String).</i> |
| spanColumnMinSpaceBefore | <code>variant spanColumnMinSpaceBefore</code> | <i>The minimum space before a span or a split column. Type: Unit (Double or String).</i> |
| spanColumnType | <code>#SpanColumnTypeOptions spanColumnType</code> | <i>Whether a paragraph should be a single column, span columns or split columns.</i> |
| spanSplitColumnCount | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator.</i> |

| TEXTSTYLERANGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--------------------------------------|--|
| splineItems | {SplineItems} *splineItems | The spline items collection. |
| splitColumnInsideGutter | variant splitColumnInsideGutter | The inside gutter if the paragraph splits columns. Type: Unit (Double or String). |
| splitColumnOutsideGutter | variant splitColumnOutsideGutter | The outside gutter if the paragraph splits columns. Type: Unit (Double or String). |
| tatechuyoko | boolean tatechuyoko | If true, makes the character horizontal in vertical text. |
| tatechuyokoXOffset | double tatechuyokoXOffset | The horizontal offset for horizontal characters in vertical text. |
| tatechuyokoYOffset | double tatechuyokoYOffset | The vertical offset for horizontal characters in vertical text. |
| trailingAki | double trailingAki | The amount of space after each character. |
| treatIdeographicSpaceAsSpace | boolean treatIdeographicSpaceAsSpace | If true, ideographic spaces will not wrap to the next line like text characters. |
| tsume | double tsume | The amount of horizontal character compression. |
| warichu | boolean warichu | If true, turns on warichu. |
| warichuAlignment | #WarichuAlignment warichuAlignment | The warichu alignment. |
| warichuCharsAfterBreak | const warichuCharsAfterBreak | The minimum number of characters allowed after a line break. |
| warichuCharsBeforeBreak | const warichuCharsBeforeBreak | The minimum number of characters allowed before a line break. |
| warichuLines | const warichuLines | The number of lines of warichu within a single normal line. |
| warichuLineSpacing | double warichuLineSpacing | The gap between lines of warichu characters. |
| warichuSize | double warichuSize | The amount (as a percentage) to scale parent text size to determine warichu size. |

TextWrapPreference

Object available in ID 3, 4, 5, 6, 7

| TEXTWRAPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|----------------------------|--------------------------------------|
| preferences | {Preferences} *preferences | A collection of preferences objects. |

ViewPreference

Object available in ID 3, 4, 5, 6, 7

| VIEWPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|---|--|
| printDialogMeasurementUnits | #MeasurementUnits printDialogMeasurementUnits | The measurement unit for the print dialog. |
| strokeMeasurementUnits | #MeasurementUnits strokeMeasurementUnits | The measurement unit for stroke measurements. |
| textSizeMeasurementUnits | #MeasurementUnits textSizeMeasurementUnits | The measurement unit for text size measurements. |
| typographicMeasurementUnits | #MeasurementUnits typographicMeasurementUnits | The measurement units for typography. |

WMF

Object available in ID 3, 4, 5, 6, 7

| WMF CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|--|
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| clearTransformations | <code>void clearTransformations()</code> | Clears transformations from the WMF. Transformations include rotation, scaling, flipping, fitting, and shearing. |
| convertShape | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | Converts the WMF to a different shape. / Given: The WMF's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0). Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points). |
| select | <code>void select(#SelectionOptions ExistingSelection)</code> | Selects the object. / ExistingSelection: The selection status of the WMF in relation to previously selected objects. |
| allGraphics | <code>{Objects} *allGraphics</code> | Lists all graphics contained by the WMF. |
| allPageItems | <code>{Objects} *allPageItems</code> | Lists all page items contained by the WMF. |
| animationSettings | <code>{AnimationSetting} *animationSettings</code> | The page item animation settings. |
| bottomLeftCornerOption | <code>#CornerOptions bottomLeftCornerOption</code> | The shape to apply to the bottom left corner of rectangular shapes. |
| bottomLeftCornerRadius | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| bottomRightCornerOption | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| bottomRightCornerRadius | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| endCap | <code>#EndCap endCap</code> | The end shape of an open path. |
| endJoin | <code>#EndJoin endJoin</code> | The corner join applied to the WMF. |
| fillColor | <code>variant fillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of the WMF. Type: Swatch or String. |
| fillTint | <code>double fillTint</code> | The percent of tint to use in the WMF's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| gapColor | <code>{Swatch} gapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type. |
| gapTint | <code>double gapTint</code> | The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| gradientFillAngle | <code>double gradientFillAngle</code> | The angle of a linear gradient applied to the fill of the WMF. (Range: -180 to 180). |

| WMF CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------|--|---|
| gradientFillLength | variant gradientFillLength | The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the WMF. Type: Unit (Double or String). |
| gradientFillStart | variant gradientFillStart | The starting point (in page coordinates) of a gradient applied to the fill of the WMF, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| gradientStrokeAngle | double gradientStrokeAngle | The angle of a linear gradient applied to the stroke of the WMF. (Range: -180 to 180). |
| gradientStrokeLength | variant gradientStrokeLength | The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the WMF. Type: Unit (Double or String). |
| gradientStrokeStart | variant gradientStrokeStart | The starting point (in page coordinates) of a gradient applied to the stroke of the WMF, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| itemLayer | {Layer} itemLayer | The layer that the WMF is on. |
| leftLineEnd | #ArrowHead leftLineEnd | The arrowhead applied to the start of the path. |
| locked | boolean locked | If true, the WMF is locked. |
| miterLimit | double miterLimit | The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500). |
| name | string name | The name of the WMF. |
| overprintFill | boolean overprintFill | If true, the WMF's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors. |
| overprintGap | boolean overprintGap | If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors. |
| overprintStroke | boolean overprintStroke | If true, the WMF's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors. |
| parentPage | {Page} *parentPage | The page on which this page item appears. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |
| rightLineEnd | #ArrowHead rightLineEnd | The arrowhead applied to the end of the path. |
| strokeAlignment | #StrokeAlignment strokeAlignment | The stroke alignment applied to the WMF. |
| strokeColor | variant strokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the WMF. Type: Swatch or String. |
| strokeCornerAdjustment | #StrokeCornerAdjustment strokeCornerAdjustment | The corner adjustment applied to the WMF. |
| strokeDashAndGap | variant strokeDashAndGap | The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings). |
| strokeTint | double strokeTint | The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| strokeType | variant strokeType | The name of the stroke style to apply. Type: StrokeStyle or String. |
| strokeWeight | variant strokeWeight | The weight (in points) to apply to the WMF's stroke. Type: Unit (Double or String). |
| timingSettings | {TimingSetting} *timingSettings | The object timing settings. |
| topLeftCornerOption | #CornerOptions topLeftCornerOption | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. |

| WMF CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|---|
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes.</i> Type: Unit (Double or String). |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes.</i> Type: Unit (Double or String). |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the WMF is visible.</i> |

Word

Object available in ID 3, 4, 5, 6, 7

| WORD CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file.</i> / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| <code>changeTransliterate</code> | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | <i>Finds text that matches the find character type value and replaces the text with the change character type value.</i> / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| <code>findHyperlinks</code> | <code>{Objects} findHyperlinks(variant SortOrder)</code> | <i>Finds hyperlink sources that intersecting with specified text range.</i> / Return value: The found hyperlinks. / SortOrder: The sort order of found ranges. Type: idRangeSortOrder enumerator. |
| <code>findTransliterate</code> | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | <i>Finds text that matches the find character type value.</i> / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| <code>showText</code> | <code>void showText()</code> | <i>Jump to the text range.</i> |
| <code>allowArbitraryHyphenation</code> | <code>boolean allowArbitraryHyphenation</code> | <i>If true, words unassociated with a hyphenation dictionary can break to the next line on any character.</i> |
| <code>autoTcy</code> | <code>const autoTcy</code> | <i>The number of half-width characters at or below which the characters automatically run horizontally in vertical text.</i> |
| <code>autoTcyIncludeRoman</code> | <code>boolean autoTcyIncludeRoman</code> | <i>If true, auto tcy includes Roman characters.</i> |
| <code>bunriKinshi</code> | <code>boolean bunriKinshi</code> | <i>If true, adds the double period (。), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect.</i> |
| <code>characterAlignment</code> | <code>#CharacterAlignment characterAlignment</code> | <i>The alignment of small characters to the largest character in the line.</i> |
| <code>characterRotation</code> | <code>double characterRotation</code> | <i>The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise.</i> |
| <code>cjkGridTracking</code> | <code>boolean cjkGridTracking</code> | <i>If true, uses grid tracking to track non-Roman characters in CJK grids.</i> |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |
| <code>glyphForm</code> | <code>#AlternateGlyphForms glyphForm</code> | <i>The glyph variant to substitute for standard glyphs.</i> |

| WORD CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|--|
| <code>gridAlignment</code> | <code>#GridAlignment gridAlignment</code> | <i>The alignment to the frame grid or baseline grid.</i> |
| <code>gridGyoudori</code> | <code>const gridGyoudori</code> | <i>The manual gyoudori setting.</i> |
| <code>jidori</code> | <code>const jidori</code> | <i>The number of grid squares in which to arrange the text.</i> |
| <code>keepWithPrevious</code> | <code>boolean keepWithPrevious</code> | <i>If the first line in the paragraph should be kept with the last line of previous paragraph.</i> |
| <code>kentenAlignment</code> | <code>#KentenAlignment kentenAlignment</code> | <i>The alignment of kenten characters relative to the parent characters.</i> |
| <code>kentenCharacterSet</code> | <code>#KentenCharacterSet kentenCharacterSet</code> | <i>The character set used for the custom kenten character. Note: Valid only when kenten kind is custom.</i> |
| <code>kentenCustomCharacter</code> | <code>string kentenCustomCharacter</code> | <i>The character used for kenten. Note: Valid only when kenten kind is custom.</i> |
| <code>kentenFillColor</code> | <code>variant kentenFillColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch or String.</i> |
| <code>kentenFont</code> | <code>variant kentenFont</code> | <i>The font to use for kenten characters. Type: Font or String.</i> |
| <code>kentenFontSize</code> | <code>double kentenFontSize</code> | <i>The size (in points) of kenten characters.</i> |
| <code>kentenFontStyle</code> | <code>variant kentenFontStyle</code> | <i>The font style of kenten characters. Type: String or idNothingEnum enumerator.</i> |
| <code>kentenKind</code> | <code>#KentenCharacter kentenKind</code> | <i>The style of kenten characters.</i> |
| <code>kentenOverprintFill</code> | <code>#AdornmentOverprint kentenOverprintFill</code> | <i>The method of overprinting the kenten fill.</i> |
| <code>kentenOverprintStroke</code> | <code>#AdornmentOverprint kentenOverprintStroke</code> | <i>The method of overprinting the kenten stroke.</i> |
| <code>kentenPlacement</code> | <code>double kentenPlacement</code> | <i>The distance between kenten characters and their parent characters.</i> |
| <code>kentenPosition</code> | <code>#RubyKetenPosition kentenPosition</code> | <i>The kenten position relative to the parent character.</i> |
| <code>kentenStrokeColor</code> | <code>variant kentenStrokeColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of kenten characters. Type: Swatch or String.</i> |
| <code>kentenStrokeTint</code> | <code>double kentenStrokeTint</code> | <i>The stroke tint (as a percentage) of kenten characters. (Range: 0 to 100).</i> |
| <code>kentenTint</code> | <code>double kentenTint</code> | <i>The fill tint (as a percentage) of kenten characters. (Range: 0 to 100).</i> |
| <code>kentenWeight</code> | <code>double kentenWeight</code> | <i>The stroke weight (in points) of kenten characters.</i> |
| <code>kentenXScale</code> | <code>double kentenXScale</code> | <i>The horizontal size of kenten characters as a percent of the original size.</i> |
| <code>kentenYScale</code> | <code>double kentenYScale</code> | <i>The vertical size of kenten characters as a percent of the original size.</i> |
| <code>kinsokuHangType</code> | <code>#KinsokuHangTypes kinsokuHangType</code> | <i>The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect.</i> |
| <code>kinsokuSet</code> | <code>variant kinsokuSet</code> | <i>The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator or String.</i> |
| <code>kinsokuType</code> | <code>#KinsokuType kinsokuType</code> | <i>The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined.</i> |
| <code>leadingAki</code> | <code>double leadingAki</code> | <i>The amount of space before each character.</i> |
| <code>leadingModel</code> | <code>#LeadingModel leadingModel</code> | <i>The point from which leading is measured from line to line.</i> |
| <code>mojikumi</code> | <code>variant mojikumi</code> | <i>The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String or idMojikumiTableDefaults enumerator.</i> |
| <code>multiStateObjects</code> | <code>{MultiStateObjects} •multiStateObjects</code> | <i>A collection of multi-state objects.</i> |

| WORD CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>otfHVKana</code> | <code>boolean otfHVKana</code> | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. |
| <code>otfProportionalMetrics</code> | <code>boolean otfProportionalMetrics</code> | If true, kerns according to proportional CJK metrics in OpenType fonts. |
| <code>otfRomanItalics</code> | <code>boolean otfRomanItalics</code> | If true, applies italics to half-width alphanumerics. |
| <code>paragraphGyoudori</code> | <code>boolean paragraphGyoudori</code> | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| <code>rensuuji</code> | <code>boolean renssuuji</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. |
| <code>rotateSingleByteCharacters</code> | <code>boolean rotateSingleByteCharacters</code> | If true, rotates Roman characters in vertical text. |
| <code>rubyAlignment</code> | <code>#RubyAlignments rubyAlignment</code> | The ruby alignment. |
| <code>rubyAutoAlign</code> | <code>boolean rubyAutoAlign</code> | If true, auto aligns ruby. |
| <code>rubyAutoScaling</code> | <code>boolean rubyAutoScaling</code> | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. |
| <code>rubyAutoTcyAutoSize</code> | <code>boolean rubyAutoTcyAutoSize</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. |
| <code>rubyAutoTcyDigits</code> | <code>const rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. |
| <code>rubyAutoTcyIncludeRoman</code> | <code>boolean rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. |
| <code>rubyFill</code> | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch or String. |
| <code>rubyFlag</code> | <code>boolean rubyFlag</code> | If true, ruby is on. |
| <code>rubyFont</code> | <code>variant rubyFont</code> | The font applied to ruby characters. Type: Font or String. |
| <code>rubyFontSize</code> | <code>double rubyFontSize</code> | The size (in points) of ruby characters. |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: String or idNothingEnum enumerator. |
| <code>rubyOpenTypePro</code> | <code>boolean rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. |
| <code>rubyOverhang</code> | <code>boolean rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. |
| <code>rubyOverprintFill</code> | <code>#AdornmentOverprint rubyOverprintFill</code> | The method of overprinting the ruby fill. |
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | The method of overprinting the ruby stroke. |
| <code>rubyParentOverhangAmount</code> | <code>#RubyOverhang rubyParentOverhangAmount</code> | The amount by which ruby characters can overhang the parent text. |
| <code>rubyParentScalingPercent</code> | <code>double rubyParentScalingPercent</code> | The amount (as a percentage) to scale the parent text size to determine the ruby text size. |
| <code>rubyParentSpacing</code> | <code>#RubyParentSpacing rubyParentSpacing</code> | The ruby spacing relative to the parent text. |
| <code>rubyPosition</code> | <code>#RubyKetenPosition rubyPosition</code> | The position of ruby characters relative to the parent text. |
| <code>rubyString</code> | <code>string rubyString</code> | The ruby string contents. |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String. |
| <code>rubyStrokeTint</code> | <code>double rubyStrokeTint</code> | The stroke tint (as a percentage) of ruby characters. |

| WORD CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| <code>rubyTint</code> | <code>double rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100).</i> |
| <code>rubyType</code> | <code>#RubyTypes rubyType</code> | <i>The ruby type.</i> |
| <code>rubyWeight</code> | <code>double rubyWeight</code> | <i>The stroke weight (in points) of ruby characters.</i> |
| <code>rubyXOffset</code> | <code>double rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters.</i> |
| <code>rubyXScale</code> | <code>double rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size.</i> |
| <code>rubyYOffset</code> | <code>double rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters.</i> |
| <code>rubyYScale</code> | <code>double rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size.</i> |
| <code>scaleAffectsLineHeight</code> | <code>boolean scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled.</i> |
| <code>shataiAdjustRotation</code> | <code>boolean shataiAdjustRotation</code> | <i>If true, applies shatai rotation.</i> |
| <code>shataiAdjustTsume</code> | <code>boolean shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume.</i> |
| <code>shataiDegreeAngle</code> | <code>double shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees).</i> |
| <code>shataiMagnification</code> | <code>double shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply.</i> |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | <i>The minimum space after a span or a split column. Type: Unit (Double or String).</i> |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | <i>The minimum space before a span or a split column. Type: Unit (Double or String).</i> |
| <code>spanColumnType</code> | <code>#SpanColumnTypeOptions spanColumnType</code> | <i>Whether a paragraph should be a single column, span columns or split columns.</i> |
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator.</i> |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | <i>The inside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>tatechuyoko</code> | <code>boolean tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text.</i> |
| <code>tatechuyokoXOffset</code> | <code>double tatechuyokoXOffset</code> | <i>The horizontal offset for horizontal characters in vertical text.</i> |
| <code>tatechuyokoYOffset</code> | <code>double tatechuyokoYOffset</code> | <i>The vertical offset for horizontal characters in vertical text.</i> |
| <code>trailingAki</code> | <code>double trailingAki</code> | <i>The amount of space after each character.</i> |
| <code>treatIdeographicSpaceAsSpace</code> | <code>boolean treatIdeographicSpaceAsSpace</code> | <i>If true, ideographic spaces will not wrap to the next line like text characters.</i> |
| <code>tsume</code> | <code>double tsume</code> | <i>The amount of horizontal character compression.</i> |
| <code>warichu</code> | <code>boolean warichu</code> | <i>If true, turns on warichu.</i> |
| <code>warichuAlignment</code> | <code>#WarichuAlignment warichuAlignment</code> | <i>The warichu alignment.</i> |
| <code>warichuCharsAfterBreak</code> | <code>const warichuCharsAfterBreak</code> | <i>The minimum number of characters allowed after a line break.</i> |
| <code>warichuCharsBeforeBreak</code> | <code>const warichuCharsBeforeBreak</code> | <i>The minimum number of characters allowed before a line break.</i> |
| <code>warichuLines</code> | <code>const warichuLines</code> | <i>The number of lines of warichu within a single normal line.</i> |

| WORD CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|--|--|
| <code>warichuLineSpacing</code> | <code>double warichuLineSpacing</code> | <i>The gap between lines of warichu characters.</i> |
| <code>warichuSize</code> | <code>double warichuSize</code> | <i>The amount (as a percentage) to scale parent text size to determine warichu size.</i> |

XMLElement

Object available in ID 3, 4, 5, 6, 7

| XMLELEMENT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| <code>changeTransliterate</code> | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | <i>Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean.</i> |
| <code>findTransliterate</code> | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | <i>Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean.</i> |
| <code>graphics</code> | <code>{Graphics} *graphics</code> | <i>A collection of imported graphics in any graphic file format (vector, metafile, or bitmap).</i> |

DataMerge

Object available in ID 4, 5, 6, 7

| DATAMERGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|---|---|
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |

Footnote

Object available in ID 4, 5, 6, 7

| FOOTNOTE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|---|---|
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |
| <code>name</code> | <code>string name</code> | <i>The name of the Footnote; this is an alias to the Footnote's label property.</i> |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |

Note

Object available in ID 4, 5, 6, 7

| NOTE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------|--------------------------|--|
| <code>name</code> | <code>string name</code> | The name of the Note; this is an alias to the Note's label property. |

ObjectStyle

Object available in ID 4, 5, 6, 7

| OBJECTSTYLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|---|---|
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | The shape to apply to the bottom left corner of rectangular shapes. |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String). |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | The shape to apply to the top right corner of rectangular shapes. |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String). |

ScriptPreference

Object available in ID 4, 5, 6, 7

| SCRIPTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--------------------------------------|---|
| <code>measurementUnit</code> | <code>variant measurementUnit</code> | The measurement unit used during script processing. Type: idAutoEnum enumerator or idMeasurementUnits enumerator. |

ChangeGrepPreference

Object available in ID 5, 6, 7

| CHANGEGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|-----------------------------------|---|
| allowArbitraryHyphenation | variant allowArbitraryHyphenation | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. Type: Boolean or idNothingEnum enumerator. |
| autoTcy | variant autoTcy | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. Type: Integer or idNothingEnum enumerator. |
| autoTcyIncludeRoman | variant autoTcyIncludeRoman | If true, auto tcy includes Roman characters. Type: Boolean or idNothingEnum enumerator. |
| bunriKinshi | variant bunriKinshi | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. Type: Boolean or idNothingEnum enumerator. |
| characterAlignment | variant characterAlignment | The alignment of small characters to the largest character in the line. Type: idCharacterAlignment enumerator or idNothingEnum enumerator. |
| characterRotation | variant characterRotation | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. Type: Double or idNothingEnum enumerator. |
| cjkGridTracking | variant cjkGridTracking | If true, uses grid tracking to track non-Roman characters in CJK grids. Type: Boolean or idNothingEnum enumerator. |
| glyphForm | variant glyphForm | The glyph variant to substitute for standard glyphs. Type: idAlternateGlyphForms enumerator or idNothingEnum enumerator. |
| gridAlignment | variant gridAlignment | The alignment to the frame grid or baseline grid. Type: idGridAlignment enumerator or idNothingEnum enumerator. |
| gridGyoudori | variant gridGyoudori | The manual gyoudori setting. Type: Integer or idNothingEnum enumerator. |
| jidori | variant jidori | The number of grid squares in which to arrange the text. . Type: Integer or idNothingEnum enumerator. |
| keepWithPrevious | variant keepWithPrevious | If the first line in the paragraph should be kept with the last line of previous paragraph. Type: Boolean or idNothingEnum enumerator. |
| kntenAlignment | variant kntenAlignment | The alignment of knten characters relative to the parent characters. . Type: idKntenAlignment enumerator or idNothingEnum enumerator. |
| kntenCharacterSet | variant kntenCharacterSet | The character set used for the custom knten character. Note: Valid only when knten kind is custom. . Type: idKntenCharacterSet enumerator or idNothingEnum enumerator. |
| kntenCustomCharacter | variant kntenCustomCharacter | The character used for knten. Note: Valid only when knten kind is custom. Type: String or idNothingEnum enumerator. |
| kntenFillColor | variant kntenFillColor | The swatch (color, gradient, tint, or mixed ink) applied to the fill of knten characters. Type: Swatch, String or idNothingEnum enumerator. |
| kntenFont | variant kntenFont | The font to use for knten characters. Type: Font, String or idNothingEnum enumerator. |
| kntenFontSize | variant kntenFontSize | The size (in points) of knten characters. Type: Double or idNothingEnum enumerator. |
| kntenFontStyle | variant kntenFontStyle | The font style of knten characters. Type: String or idNothingEnum enumerator. |
| kntenKind | variant kntenKind | The style of knten characters. Type: idKntenCharacter enumerator or idNothingEnum enumerator. |

| CHANGEGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--------------------------------|---|
| kntenOverprintFill | variant kntenOverprintFill | The method of overprinting the knten fill. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| kntenOverprintStroke | variant kntenOverprintStroke | The method of overprinting the knten stroke. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| kntenPlacement | variant kntenPlacement | The distance between knten characters and their parent characters. Type: <code>Double</code> or <code>idNothingEnum</code> enumerator. |
| kntenPosition | variant kntenPosition | The knten position relative to the parent character. Type: <code>idRubyKetenPosition</code> enumerator or <code>idNothingEnum</code> enumerator. |
| kntenStrokeColor | variant kntenStrokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of knten characters. Type: <code>Swatch</code> , <code>String</code> or <code>idNothingEnum</code> enumerator. |
| kntenStrokeTint | variant kntenStrokeTint | The stroke tint (as a percentage) of knten characters. (Range: 0 to 100). Type: <code>Double</code> or <code>idNothingEnum</code> enumerator. |
| kntenTint | variant kntenTint | The fill tint (as a percentage) of knten characters. (Range: 0 to 100). Type: <code>Double</code> or <code>idNothingEnum</code> enumerator. |
| kntenWeight | variant kntenWeight | The stroke weight (in points) of knten characters. Type: <code>Double</code> or <code>idNothingEnum</code> enumerator. |
| kntenXScale | variant kntenXScale | The horizontal size of knten characters as a percent of the original size. Type: <code>Double</code> or <code>idNothingEnum</code> enumerator. |
| kntenYScale | variant kntenYScale | The vertical size of knten characters as a percent of the original size. . Type: <code>Double</code> or <code>idNothingEnum</code> enumerator. |
| kinsokuHangType | variant kinsokuHangType | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. Type: <code>idKinsokuHangTypes</code> enumerator or <code>idNothingEnum</code> enumerator. |
| kinsokuSet | variant kinsokuSet | The kinsoku set that determines legitimate line breaks. Type: <code>KinsokuTable</code> , <code>idKinsokuSet</code> enumerator, <code>String</code> or <code>idNothingEnum</code> enumerator. |
| kinsokuType | variant kinsokuType | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. Type: <code>idKinsokuType</code> enumerator or <code>idNothingEnum</code> enumerator. |
| leadingAki | variant leadingAki | The amount of space before each character. Type: <code>Double</code> or <code>idNothingEnum</code> enumerator. |
| leadingModel | variant leadingModel | The point from which leading is measured from line to line. Type: <code>idLeadingModel</code> enumerator or <code>idNothingEnum</code> enumerator. |
| mojikumi | variant mojikumi | The mojikumi table. For information, see mojikumi table defaults. Type: <code>MojikumiTable</code> , <code>String</code> , <code>idMojikumiTableDefaults</code> enumerator or <code>idNothingEnum</code> enumerator. |
| otfHVKana | variant otfHVKana | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. . Type: <code>Boolean</code> or <code>idNothingEnum</code> enumerator. |
| otfProportionalMetrics | variant otfProportionalMetrics | If true, kerns according to proportional CJK metrics in OpenType fonts. Type: <code>Boolean</code> or <code>idNothingEnum</code> enumerator. |
| otfRomanItalics | variant otfRomanItalics | If true, applies italics to half-width alphanumerics. . Type: <code>Boolean</code> or <code>idNothingEnum</code> enumerator. |
| paragraphGyoudori | variant paragraphGyoudori | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. Type: <code>Boolean</code> or <code>idNothingEnum</code> enumerator. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |

| CHANGEGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| <code>rensuumi</code> | <code>variant rensuumi</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rotateSingleByteCharacters</code> | <code>variant rotateSingleByteCharacters</code> | If true, rotates Roman characters in vertical text. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAlignment</code> | <code>variant rubyAlignment</code> | The ruby alignment. Type: <code>idRubyAlignments</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoAlign</code> | <code>variant rubyAutoAlign</code> | If true, auto aligns ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoScaling</code> | <code>variant rubyAutoScaling</code> | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see <code>ruby parent scaling percent</code> . . Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyAutoScale</code> | <code>variant rubyAutoTcyAutoScale</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyDigits</code> | <code>variant rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. Type: Integer or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyIncludeRoman</code> | <code>variant rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyFill</code> | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyFont</code> | <code>variant rubyFont</code> | The font applied to ruby characters. Type: Font, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyFontSize</code> | <code>variant rubyFontSize</code> | The size (in points) of ruby characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: String or <code>idNothingEnum</code> enumerator. |
| <code>rubyOpenTypePro</code> | <code>variant rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverhang</code> | <code>variant rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see <code>ruby parent overhang amount</code> . . Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverprintFill</code> | <code>variant rubyOverprintFill</code> | The method of overprinting the ruby fill. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverprintStroke</code> | <code>variant rubyOverprintStroke</code> | The method of overprinting the ruby stroke. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentOverhangAmount</code> | <code>variant rubyParentOverhangAmount</code> | The amount by which ruby characters can overhang the parent text. Type: <code>idRubyOverhang</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentScalingPercent</code> | <code>variant rubyParentScalingPercent</code> | The amount (as a percentage) to scale the parent text size to determine the ruby text size. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentSpacing</code> | <code>variant rubyParentSpacing</code> | The ruby spacing relative to the parent text. . Type: <code>idRubyParentSpacing</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyPosition</code> | <code>variant rubyPosition</code> | The position of ruby characters relative to the parent text. Type: <code>idRubyKertenPosition</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyStrokeTint</code> | <code>variant rubyStrokeTint</code> | The stroke tint (as a percentage) of ruby characters. Type: Double or <code>idNothingEnum</code> enumerator. |

| CHANGEGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|---|---|
| <code>rubyTint</code> | <code>variant rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). Type: Double or idNothingEnum enumerator.</i> |
| <code>rubyType</code> | <code>variant rubyType</code> | <i>The ruby type. Type: idRubyTypes enumerator or idNothingEnum enumerator.</i> |
| <code>rubyWeight</code> | <code>variant rubyWeight</code> | <i>The stroke weight (in points) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| <code>rubyXOffset</code> | <code>variant rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| <code>rubyXScale</code> | <code>variant rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| <code>rubyYOffset</code> | <code>variant rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| <code>rubyYScale</code> | <code>variant rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| <code>scaleAffectsLineHeight</code> | <code>variant scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled. . Type: Boolean or idNothingEnum enumerator.</i> |
| <code>shataiAdjustRotation</code> | <code>variant shataiAdjustRotation</code> | <i>If true, applies shatai rotation. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>shataiAdjustTsume</code> | <code>variant shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>shataiDegreeAngle</code> | <code>variant shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees). Type: Double or idNothingEnum enumerator.</i> |
| <code>shataiMagnification</code> | <code>variant shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply. Type: Double or idNothingEnum enumerator.</i> |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | <i>The minimum space after a span or a split column. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | <i>The minimum space before a span or a split column. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>spanColumnType</code> | <code>variant spanColumnType</code> | <i>Whether a paragraph should be a single column, span columns or split columns. Type: idSpanColumnTypeOptions enumerator or idNothingEnum enumerator.</i> |
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40), idSpanColumnCountOptions enumerator or idNothingEnum enumerator.</i> |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | <i>The inside gutter if the paragraph splits columns. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>tatechuyoko</code> | <code>variant tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>tatechuyokoXOffset</code> | <code>variant tatechuyokoXOffset</code> | <i>The horizontal offset for horizontal characters in vertical text. Type: Double or idNothingEnum enumerator.</i> |
| <code>tatechuyokoYOffset</code> | <code>variant tatechuyokoYOffset</code> | <i>The vertical offset for horizontal characters in vertical text. Type: Double or idNothingEnum enumerator.</i> |
| <code>trailingAki</code> | <code>variant trailingAki</code> | <i>The amount of space after each character. Type: Double or idNothingEnum enumerator.</i> |

| CHANGEGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--------------------------------------|---|
| treatIdeographicSpaceAsSpace | variant treatIdeographicSpaceAsSpace | If true, ideographic spaces will not wrap to the next line like text characters. Type: Boolean or idNothingEnum enumerator. |
| tsume | variant tsume | The amount of horizontal character compression. Type: Double or idNothingEnum enumerator. |
| warichu | variant warichu | If true, turns on warichu. Type: Boolean or idNothingEnum enumerator. |
| warichuAlignment | variant warichuAlignment | The warichu alignment. Type: idWarichuAlignment enumerator or idNothingEnum enumerator. |
| warichuCharsAfterBreak | variant warichuCharsAfterBreak | The minimum number of characters allowed after a line break. . Type: Integer or idNothingEnum enumerator. |
| warichuCharsBeforeBreak | variant warichuCharsBeforeBreak | The minimum number of characters allowed before a line break. Type: Integer or idNothingEnum enumerator. |
| warichuLines | variant warichuLines | The number of lines of warichu within a single normal line. Type: Integer or idNothingEnum enumerator. |
| warichuLineSpacing | variant warichuLineSpacing | The gap between lines of warichu characters. Type: Double or idNothingEnum enumerator. |
| warichuSize | variant warichuSize | The amount (as a percentage) to scale parent text size to determine warichu size. Type: Double or idNothingEnum enumerator. |

ChangeObjectPreference

Object available in ID 5, 6, 7

| CHANGEOBJECTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|---------------------------------|---|
| bottomLeftCornerOption | variant bottomLeftCornerOption | The shape to apply to the bottom left corner of rectangular shapes. Type: idCornerOptions enumerator or idNothingEnum enumerator. |
| bottomLeftCornerRadius | variant bottomLeftCornerRadius | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String) or idNothingEnum enumerator. |
| bottomRightCornerOption | variant bottomRightCornerOption | The shape to apply to the bottom right corner of rectangular shapes. Type: idCornerOptions enumerator or idNothingEnum enumerator. |
| bottomRightCornerRadius | variant bottomRightCornerRadius | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String) or idNothingEnum enumerator. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |
| topLeftCornerOption | variant topLeftCornerOption | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. Type: idCornerOptions enumerator or idNothingEnum enumerator. |
| topLeftCornerRadius | variant topLeftCornerRadius | The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String) or idNothingEnum enumerator. |
| topRightCornerOption | variant topRightCornerOption | The shape to apply to the top right corner of rectangular shapes. Type: idCornerOptions enumerator or idNothingEnum enumerator. |

| CHANGEOBJECTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|---|--|
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String) or idNothingEnum enumerator. |

ChangeTextPreference

Object available in ID 5, 6, 7

| CHANGETEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|---|
| <code>allowArbitraryHyphenation</code> | <code>variant allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. Type: Boolean or idNothingEnum enumerator. |
| <code>autoTcy</code> | <code>variant autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. Type: Integer or idNothingEnum enumerator. |
| <code>autoTcyIncludeRoman</code> | <code>variant autoTcyIncludeRoman</code> | If true, auto tcy includes Roman characters. Type: Boolean or idNothingEnum enumerator. |
| <code>bunriKinshi</code> | <code>variant bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. Type: Boolean or idNothingEnum enumerator. |
| <code>characterAlignment</code> | <code>variant characterAlignment</code> | The alignment of small characters to the largest character in the line. Type: idCharacterAlignment enumerator or idNothingEnum enumerator. |
| <code>characterRotation</code> | <code>variant characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. Type: Double or idNothingEnum enumerator. |
| <code>cjkGridTracking</code> | <code>variant cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. Type: Boolean or idNothingEnum enumerator. |
| <code>glyphForm</code> | <code>variant glyphForm</code> | The glyph variant to substitute for standard glyphs. Type: idAlternateGlyphForms enumerator or idNothingEnum enumerator. |
| <code>gridAlignment</code> | <code>variant gridAlignment</code> | The alignment to the frame grid or baseline grid. Type: idGridAlignment enumerator or idNothingEnum enumerator. |
| <code>gridGyoudori</code> | <code>variant gridGyoudori</code> | The manual gyoudori setting. Type: Integer or idNothingEnum enumerator. |
| <code>jidori</code> | <code>variant jidori</code> | The number of grid squares in which to arrange the text. Type: Integer or idNothingEnum enumerator. |
| <code>keepWithPrevious</code> | <code>variant keepWithPrevious</code> | If the first line in the paragraph should be kept with the last line of previous paragraph. Type: Boolean or idNothingEnum enumerator. |
| <code>kentenAlignment</code> | <code>variant kentenAlignment</code> | The alignment of kenten characters relative to the parent characters. Type: idKentenAlignment enumerator or idNothingEnum enumerator. |
| <code>kentenCharacterSet</code> | <code>variant kentenCharacterSet</code> | The character set used for the custom kenten character. Note: Valid only when kenten kind is custom. Type: idKentenCharacterSet enumerator or idNothingEnum enumerator. |
| <code>kentenCustomCharacter</code> | <code>variant kentenCustomCharacter</code> | The character used for kenten. Note: Valid only when kenten kind is custom. Type: String or idNothingEnum enumerator. |
| <code>kentenFillColor</code> | <code>variant kentenFillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch, String or idNothingEnum enumerator. |
| <code>kentenFont</code> | <code>variant kentenFont</code> | The font to use for kenten characters. Type: Font, String or idNothingEnum enumerator. |

| CHANGETEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--------------------------------|---|
| kntenFontSize | variant kntenFontSize | The size (in points) of knten characters. Type: Double or idNothingEnum enumerator. |
| kntenFontStyle | variant kntenFontStyle | The font style of knten characters. Type: String or idNothingEnum enumerator. |
| kntenKind | variant kntenKind | The style of knten characters. Type: idKntenCharacter enumerator or idNothingEnum enumerator. |
| kntenOverprintFill | variant kntenOverprintFill | The method of overprinting the knten fill. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator. |
| kntenOverprintStroke | variant kntenOverprintStroke | The method of overprinting the knten stroke. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator. |
| kntenPlacement | variant kntenPlacement | The distance between knten characters and their parent characters. Type: Double or idNothingEnum enumerator. |
| kntenPosition | variant kntenPosition | The knten position relative to the parent character. Type: idRubyKetenPosition enumerator or idNothingEnum enumerator. |
| kntenStrokeColor | variant kntenStrokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of knten characters. Type: Swatch, String or idNothingEnum enumerator. |
| kntenStrokeTint | variant kntenStrokeTint | The stroke tint (as a percentage) of knten characters. (Range: 0 to 100). Type: Double or idNothingEnum enumerator. |
| kntenTint | variant kntenTint | The fill tint (as a percentage) of knten characters. (Range: 0 to 100). Type: Double or idNothingEnum enumerator. |
| kntenWeight | variant kntenWeight | The stroke weight (in points) of knten characters. Type: Double or idNothingEnum enumerator. |
| kntenXScale | variant kntenXScale | The horizontal size of knten characters as a percent of the original size. Type: Double or idNothingEnum enumerator. |
| kntenYScale | variant kntenYScale | The vertical size of knen characters as a percent of the original size. . Type: Double or idNothingEnum enumerator. |
| kinsokuHangType | variant kinsokuHangType | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. Type: idKinsokuHangTypes enumerator or idNothingEnum enumerator. |
| kinsokuSet | variant kinsokuSet | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator, String or idNothingEnum enumerator. |
| kinsokuType | variant kinsokuType | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. Type: idKinsokuType enumerator or idNothingEnum enumerator. |
| leadingAki | variant leadingAki | The amount of space before each character. Type: Double or idNothingEnum enumerator. |
| leadingModel | variant leadingModel | The point from which leading is measured from line to line. Type: idLeadingModel enumerator or idNothingEnum enumerator. |
| mojikumi | variant mojikumi | The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String, idMojikumiTableDefaults enumerator or idNothingEnum enumerator. |
| otfHVKana | variant otfHVKana | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. . Type: Boolean or idNothingEnum enumerator. |
| otfProportionalMetrics | variant otfProportionalMetrics | If true, kerns according to proportional CJK metrics in OpenType fonts. Type: Boolean or idNothingEnum enumerator. |

| CHANGETEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| <code>otfRomanItalics</code> | <code>variant otfRomanItalics</code> | If true, applies italics to half-width alphanumeric. . Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>paragraphGyoudori</code> | <code>variant paragraphGyoudori</code> | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| <code>rensuumi</code> | <code>variant rensumi</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rotateSingleByteCharacters</code> | <code>variant rotateSingleByteCharacters</code> | If true, rotates Roman characters in vertical text. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAlignment</code> | <code>variant rubyAlignment</code> | The ruby alignment. Type: <code>idRubyAlignments</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoAlign</code> | <code>variant rubyAutoAlign</code> | If true, auto aligns ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoScaling</code> | <code>variant rubyAutoScaling</code> | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. . Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyAutoSize</code> | <code>variant rubyAutoTcyAutoSize</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyDigits</code> | <code>variant rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. Type: Integer or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyIncludeRoman</code> | <code>variant rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyFill</code> | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyFont</code> | <code>variant rubyFont</code> | The font applied to ruby characters. Type: Font, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyFontSize</code> | <code>variant rubyFontSize</code> | The size (in points) of ruby characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: String or <code>idNothingEnum</code> enumerator. |
| <code>rubyOpenTypePro</code> | <code>variant rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverhang</code> | <code>variant rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. . Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverprintFill</code> | <code>variant rubyOverprintFill</code> | The method of overprinting the ruby fill. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverprintStroke</code> | <code>variant rubyOverprintStroke</code> | The method of overprinting the ruby stroke. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentOverhangAmount</code> | <code>variant rubyParentOverhangAmount</code> | The amount by which ruby characters can overhang the parent text. Type: <code>idRubyOverhang</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentScalingPercent</code> | <code>variant rubyParentScalingPercent</code> | The amount (as a percentage) to scale the parent text size to determine the ruby text size. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentSpacing</code> | <code>variant rubyParentSpacing</code> | The ruby spacing relative to the parent text. . Type: <code>idRubyParentSpacing</code> enumerator or <code>idNothingEnum</code> enumerator. |

| CHANGETEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|---|---|
| rubyPosition | <code>variant rubyPosition</code> | <i>The position of ruby characters relative to the parent text. Type: idRubyKertenPosition enumerator or idNothingEnum enumerator.</i> |
| rubyStroke | <code>variant rubyStroke</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch, String or idNothingEnum enumerator.</i> |
| rubyStrokeTint | <code>variant rubyStrokeTint</code> | <i>The stroke tint (as a percentage) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyTint | <code>variant rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). Type: Double or idNothingEnum enumerator.</i> |
| rubyType | <code>variant rubyType</code> | <i>The ruby type. Type: idRubyTypes enumerator or idNothingEnum enumerator.</i> |
| rubyWeight | <code>variant rubyWeight</code> | <i>The stroke weight (in points) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyXOffset | <code>variant rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyXScale | <code>variant rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| rubyYOffset | <code>variant rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyYScale | <code>variant rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| scaleAffectsLineHeight | <code>variant scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled. . Type: Boolean or idNothingEnum enumerator.</i> |
| shataiAdjustRotation | <code>variant shataiAdjustRotation</code> | <i>If true, applies shatai rotation. Type: Boolean or idNothingEnum enumerator.</i> |
| shataiAdjustTsume | <code>variant shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume. Type: Boolean or idNothingEnum enumerator.</i> |
| shataiDegreeAngle | <code>variant shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees). Type: Double or idNothingEnum enumerator.</i> |
| shataiMagnification | <code>variant shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply. Type: Double or idNothingEnum enumerator.</i> |
| spanColumnMinSpaceAfter | <code>variant spanColumnMinSpaceAfter</code> | <i>The minimum space after a span or a split column. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| spanColumnMinSpaceBefore | <code>variant spanColumnMinSpaceBefore</code> | <i>The minimum space before a span or a split column. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| spanColumnType | <code>variant spanColumnType</code> | <i>Whether a paragraph should be a single column, span columns or split columns. Type: idSpanColumnTypeOptions enumerator or idNothingEnum enumerator.</i> |
| spanSplitColumnCount | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40), idSpanColumnCountOptions enumerator or idNothingEnum enumerator.</i> |
| splitColumnInsideGutter | <code>variant splitColumnInsideGutter</code> | <i>The inside gutter if the paragraph splits columns. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| splitColumnOutsideGutter | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| tatechuyoko | <code>variant tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text. Type: Boolean or idNothingEnum enumerator.</i> |

| CHANGETEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|---|---|
| tatechuyokoXOffset | variant <code>tatechuyokoXOffset</code> | The horizontal offset for horizontal characters in vertical text. Type: Double or idNothingEnum enumerator. |
| tatechuyokoYOffset | variant <code>tatechuyokoYOffset</code> | The vertical offset for horizontal characters in vertical text. Type: Double or idNothingEnum enumerator. |
| trailingAki | variant <code>trailingAki</code> | The amount of space after each character. Type: Double or idNothingEnum enumerator. |
| treatIdeographicSpaceAsSpace | variant <code>treatIdeographicSpaceAsSpace</code> | If true, ideographic spaces will not wrap to the next line like text characters. Type: Boolean or idNothingEnum enumerator. |
| tsume | variant <code>tsume</code> | The amount of horizontal character compression. Type: Double or idNothingEnum enumerator. |
| warichu | variant <code>warichu</code> | If true, turns on warichu. Type: Boolean or idNothingEnum enumerator. |
| warichuAlignment | variant <code>warichuAlignment</code> | The warichu alignment. Type: idWarichuAlignment enumerator or idNothingEnum enumerator. |
| warichuCharsAfterBreak | variant <code>warichuCharsAfterBreak</code> | The minimum number of characters allowed after a line break. . Type: Integer or idNothingEnum enumerator. |
| warichuCharsBeforeBreak | variant <code>warichuCharsBeforeBreak</code> | The minimum number of characters allowed before a line break. Type: Integer or idNothingEnum enumerator. |
| warichuLines | variant <code>warichuLines</code> | The number of lines of warichu within a single normal line. Type: Integer or idNothingEnum enumerator. |
| warichuLineSpacing | variant <code>warichuLineSpacing</code> | The gap between lines of warichu characters. Type: Double or idNothingEnum enumerator. |
| warichuSize | variant <code>warichuSize</code> | The amount (as a percentage) to scale parent text size to determine warichu size. Type: Double or idNothingEnum enumerator. |

ContentTransparencySetting

Object available in ID 5, 6, 7

| CONTENTTRANSPARENCYSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|----------------------------|--------------------------------------|
| preferences | {Preferences} *preferences | A collection of preferences objects. |

EventListener

Object available in ID 5, 6, 7

| EVENTLISTENER CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|---|--|
| extractLabel | string <code>extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| insertLabel | void <code>insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| label | string <code>label</code> | A property that can be set to any string. |

| EVENTLISTENER CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--------------------------|---|
| <code>name</code> | <code>string name</code> | <i>The name of the EventListener; this is an alias to the EventListener's label property.</i> |

FillTransparencySetting

Object available in ID 5, 6, 7

| FILLTRANSPARENCYSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|---|---|
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |

FindChangeGrepOption

Object available in ID 5, 6, 7

| FINDCHANGEGREPOPTION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|------------------------------------|---|
| <code>kanaSensitive</code> | <code>boolean kanaSensitive</code> | <i>If true, finds only text that matches the specified kana type.</i> |
| <code>widthSensitive</code> | <code>boolen widthSensitive</code> | <i>If true, finds only text that matches the specified character width.</i> |

FindChangeTextOption

Object available in ID 5, 6, 7

| FINDCHANGETEXTOPTION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--------------------------------------|---|
| <code>ignoreDiacritics</code> | <code>boolen ignoreDiacritics</code> | <i>If true, ignore diacs in the find/change query.</i> |
| <code>ignoreKashidas</code> | <code>boolen ignoreKashidas</code> | <i>If true, ignore kashidas in the find/change query.</i> |
| <code>kanaSensitive</code> | <code>boolen kanaSensitive</code> | <i>If true, finds only text that matches the specified kana type.</i> |
| <code>widthSensitive</code> | <code>boolen widthSensitive</code> | <i>If true, finds only text that matches the specified character width.</i> |

FindGrepPreference

Object available in ID 5, 6, 7

| FINDGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|---|
| <code>allowArbitraryHyphenation</code> | <code>variant allowArbitraryHyphenation</code> | <i>If true, words unassociated with a hyphenation dictionary can break to the next line on any character. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>autoTcy</code> | <code>variant autoTcy</code> | <i>The number of half-width characters at or below which the characters automatically run horizontally in vertical text. Type: Integer or idNothingEnum enumerator.</i> |
| <code>autoTcyIncludeRoman</code> | <code>variant autoTcyIncludeRoman</code> | <i>If true, auto tcy includes Roman characters. Type: Boolean or idNothingEnum enumerator.</i> |

| FINDGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|-------------------------------|---|
| bunriKinshi | variant bunriKinshi | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. Type: Boolean or idNothingEnum enumerator. |
| characterAlignment | variant characterAlignment | The alignment of small characters to the largest character in the line. Type: idCharacterAlignment enumerator or idNothingEnum enumerator. |
| characterRotation | variant characterRotation | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. Type: Double or idNothingEnum enumerator. |
| cjkGridTracking | variant cjkGridTracking | If true, uses grid tracking to track non-Roman characters in CJK grids. Type: Boolean or idNothingEnum enumerator. |
| glyphForm | variant glyphForm | The glyph variant to substitute for standard glyphs. Type: idAlternateGlyphForms enumerator or idNothingEnum enumerator. |
| gridAlignment | variant gridAlignment | The alignment to the frame grid or baseline grid. Type: idGridAlignment enumerator or idNothingEnum enumerator. |
| gridGyoudori | variant gridGyoudori | The manual gyoudori setting. Type: Integer or idNothingEnum enumerator. |
| jidori | variant jidori | The number of grid squares in which to arrange the text. . Type: Integer or idNothingEnum enumerator. |
| keepWithPrevious | variant keepWithPrevious | If the first line in the paragraph should be kept with the last line of previous paragraph. Type: Boolean or idNothingEnum enumerator. |
| kentenAlignment | variant kentenAlignment | The alignment of kenten characters relative to the parent characters. . Type: idKentenAlignment enumerator or idNothingEnum enumerator. |
| kentenCharacterSet | variant kentenCharacterSet | The character set used for the custom kenten character. Note: Valid only when kenten kind is custom. . Type: idKentenCharacterSet enumerator or idNothingEnum enumerator. |
| kentenCustomCharacter | variant kentenCustomCharacter | The character used for kenten. Note: Valid only when kenten kind is custom. Type: String or idNothingEnum enumerator. |
| kentenFillColor | variant kentenFillColor | The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch, String or idNothingEnum enumerator. |
| kentenFont | variant kentenFont | The font to use for kenten characters. Type: Font, String or idNothingEnum enumerator. |
| kentenFontSize | variant kentenFontSize | The size (in points) of kenten characters. Type: Double or idNothingEnum enumerator. |
| kentenFontStyle | variant kentenFontStyle | The font style of kenten characters. Type: String or idNothingEnum enumerator. |
| kentenKind | variant kentenKind | The style of kenten characters. Type: idKentenCharacter enumerator or idNothingEnum enumerator. |
| kentenOverprintFill | variant kentenOverprintFill | The method of overprinting the kenten fill. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator. |
| kentenOverprintStroke | variant kentenOverprintStroke | The method of overprinting the kenten stroke. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator. |
| kentenPlacement | variant kentenPlacement | The distance between kenten characters and their parent characters. Type: Double or idNothingEnum enumerator. |
| kentenPosition | variant kentenPosition | The kenten position relative to the parent character. Type: idRubyKentenPosition enumerator or idNothingEnum enumerator. |
| kentenStrokeColor | variant kentenStrokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of kenten characters. Type: Swatch, String or idNothingEnum enumerator. |

| FINDGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|------------------------------------|---|
| kntenStrokeTint | variant kntenStrokeTint | The stroke tint (as a percentage) of knten characters. (Range: 0 to 100). Type: Double or idNothingEnum enumerator. |
| kntenTint | variant kntenTint | The fill tint (as a percentage) of knten characters. (Range: 0 to 100). Type: Double or idNothingEnum enumerator. |
| kntenWeight | variant kntenWeight | The stroke weight (in points) of knten characters. Type: Double or idNothingEnum enumerator. |
| kntenXScale | variant kntenXScale | The horizontal size of knten characters as a percent of the original size. Type: Double or idNothingEnum enumerator. |
| kntenYScale | variant kntenYScale | The vertical size of knten characters as a percent of the original size. . Type: Double or idNothingEnum enumerator. |
| kinsokuHangType | variant kinsokuHangType | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. Type: idKinsokuHangTypes enumerator or idNothingEnum enumerator. |
| kinsokuSet | variant kinsokuSet | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator, String or idNothingEnum enumerator. |
| kinsokuType | variant kinsokuType | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. Type: idKinsokuType enumerator or idNothingEnum enumerator. |
| leadingAki | variant leadingAki | The amount of space before each character. Type: Double or idNothingEnum enumerator. |
| leadingModel | variant leadingModel | The point from which leading is measured from line to line. Type: idLeadingModel enumerator or idNothingEnum enumerator. |
| mojikumi | variant mojikumi | The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String, idMojikumiTableDefaults enumerator or idNothingEnum enumerator. |
| otfHVKana | variant otfHVKana | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. . Type: Boolean or idNothingEnum enumerator. |
| otfProportionalMetrics | variant otfProportionalMetrics | If true, kerns according to proportional CJK metrics in OpenType fonts. Type: Boolean or idNothingEnum enumerator. |
| otfRomanItalics | variant otfRomanItalics | If true, applies italics to half-width alphanumerics. . Type: Boolean or idNothingEnum enumerator. |
| paragraphGyoudori | variant paragraphGyoudori | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. Type: Boolean or idNothingEnum enumerator. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |
| rensuuji | variant rensuuji | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. Type: Boolean or idNothingEnum enumerator. |
| rotateSingleByteCharacters | variant rotateSingleByteCharacters | If true, rotates Roman characters in vertical text. Type: Boolean or idNothingEnum enumerator. |
| rubyAlignment | variant rubyAlignment | The ruby alignment. Type: idRubyAlignments enumerator or idNothingEnum enumerator. |
| rubyAutoAlign | variant rubyAutoAlign | If true, auto aligns ruby. Type: Boolean or idNothingEnum enumerator. |
| rubyAutoScaling | variant rubyAutoScaling | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. . Type: Boolean or idNothingEnum enumerator. |

| FINDGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|---|--|
| <code>rubyAutoTcyAutoScale</code> | <code>variant rubyAutoTcyAutoScale</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyDigits</code> | <code>variant rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. Type: Integer or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyIncludeRoman</code> | <code>variant rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyFill</code> | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyFont</code> | <code>variant rubyFont</code> | The font applied to ruby characters. Type: Font, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyFontSize</code> | <code>variant rubyFontSize</code> | The size (in points) of ruby characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: String or <code>idNothingEnum</code> enumerator. |
| <code>rubyOpenTypePro</code> | <code>variant rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverhang</code> | <code>variant rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. . Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverprintFill</code> | <code>variant rubyOverprintFill</code> | The method of overprinting the ruby fill. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverprintStroke</code> | <code>variant rubyOverprintStroke</code> | The method of overprinting the ruby stroke. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentOverhangAmount</code> | <code>variant rubyParentOverhangAmount</code> | The amount by which ruby characters can overhang the parent text. Type: <code>idRubyOverhang</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentScalingPercent</code> | <code>variant rubyParentScalingPercent</code> | The amount (as a percentage) to scale the parent text size to determine the ruby text size. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentSpacing</code> | <code>variant rubyParentSpacing</code> | The ruby spacing relative to the parent text. . Type: <code>idRubyParentSpacing</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyPosition</code> | <code>variant rubyPosition</code> | The position of ruby characters relative to the parent text. Type: <code>idRubyKertenPosition</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyStrokeTint</code> | <code>variant rubyStrokeTint</code> | The stroke tint (as a percentage) of ruby characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyTint</code> | <code>variant rubyTint</code> | The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyType</code> | <code>variant rubyType</code> | The ruby type. Type: <code>idRubyTypes</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyWeight</code> | <code>variant rubyWeight</code> | The stroke weight (in points) of ruby characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyXOffset</code> | <code>variant rubyXOffset</code> | The amount of horizontal space between ruby and parent characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyXScale</code> | <code>variant rubyXScale</code> | The horizontal size of ruby characters, specified as a percent of the original size. Type: Double or <code>idNothingEnum</code> enumerator. |

| FINDGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|---|
| <code>rubyYOffset</code> | <code>variant rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| <code>rubyYScale</code> | <code>variant rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| <code>scaleAffectsLineHeight</code> | <code>variant scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled. . Type: Boolean or idNothingEnum enumerator.</i> |
| <code>shataiAdjustRotation</code> | <code>variant shataiAdjustRotation</code> | <i>If true, applies shatai rotation. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>shataiAdjustTsume</code> | <code>variant shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>shataiDegreeAngle</code> | <code>variant shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees). Type: Double or idNothingEnum enumerator.</i> |
| <code>shataiMagnification</code> | <code>variant shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply. Type: Double or idNothingEnum enumerator.</i> |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | <i>The minimum space after a span or a split column. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | <i>The minimum space before a span or a split column. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>spanColumnType</code> | <code>variant spanColumnType</code> | <i>Whether a paragraph should be a single column, span columns or split columns. Type: idSpanColumnTypeOptions enumerator or idNothingEnum enumerator.</i> |
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40), idSpanColumnCountOptions enumerator or idNothingEnum enumerator.</i> |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | <i>The inside gutter if the paragraph splits columns. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>tatechuyoko</code> | <code>variant tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>tatechuyokoXOffset</code> | <code>variant tatechuyokoXOffset</code> | <i>The horizontal offset for horizontal characters in vertical text. Type: Double or idNothingEnum enumerator.</i> |
| <code>tatechuyokoYOffset</code> | <code>variant tatechuyokoYOffset</code> | <i>The vertical offset for horizontal characters in vertical text. Type: Double or idNothingEnum enumerator.</i> |
| <code>trailingAki</code> | <code>variant trailingAki</code> | <i>The amount of space after each character. Type: Double or idNothingEnum enumerator.</i> |
| <code>treatIdeographicSpaceAsSpace</code> | <code>variant treatIdeographicSpaceAsSpace</code> | <i>If true, ideographic spaces will not wrap to the next line like text characters. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>tsume</code> | <code>variant tsume</code> | <i>The amount of horizontal character compression. Type: Double or idNothingEnum enumerator.</i> |
| <code>warichu</code> | <code>variant warichu</code> | <i>If true, turns on warichu. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>warichuAlignment</code> | <code>variant warichuAlignment</code> | <i>The warichu alignment. Type: idWarichuAlignment enumerator or idNothingEnum enumerator.</i> |
| <code>warichuCharsAfterBreak</code> | <code>variant warichuCharsAfterBreak</code> | <i>The minimum number of characters allowed after a line break. . Type: Integer or idNothingEnum enumerator.</i> |

| FINDGREPPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|--|--|
| <code>warichuCharsBeforeBreak</code> | <code>variant warichuCharsBeforeBreak</code> | <i>The minimum number of characters allowed before a line break. Type: Integer or idNothingEnum enumerator.</i> |
| <code>warichuLines</code> | <code>variant warichuLines</code> | <i>The number of lines of warichu within a single normal line. Type: Integer or idNothingEnum enumerator.</i> |
| <code>warichuLineSpacing</code> | <code>variant warichuLineSpacing</code> | <i>The gap between lines of warichu characters. Type: Double or idNothingEnum enumerator.</i> |
| <code>warichuSize</code> | <code>variant warichuSize</code> | <i>The amount (as a percentage) to scale parent text size to determine warichu size. Type: Double or idNothingEnum enumerator.</i> |

FindObjectPreference

Object available in ID 5, 6, 7

| FINDOBJECTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|--|--|
| <code>bottomLeftCornerOption</code> | <code>variant bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes. Type: idCornerOptions enumerator or idNothingEnum enumerator.</i> |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>bottomRightCornerOption</code> | <code>variant bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes. Type: idCornerOptions enumerator or idNothingEnum enumerator.</i> |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| <code>topLeftCornerOption</code> | <code>variant topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. Type: idCornerOptions enumerator or idNothingEnum enumerator.</i> |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>topRightCornerOption</code> | <code>variant topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes. Type: idCornerOptions enumerator or idNothingEnum enumerator.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |

FindTextPreference

Object available in ID 5, 6, 7

| FINDTEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|-----------------------------------|---|
| allowArbitraryHyphenation | variant allowArbitraryHyphenation | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. Type: Boolean or idNothingEnum enumerator. |
| autoTcy | variant autoTcy | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. Type: Integer or idNothingEnum enumerator. |
| autoTcyIncludeRoman | variant autoTcyIncludeRoman | If true, auto tcy includes Roman characters. Type: Boolean or idNothingEnum enumerator. |
| bunriKinshi | variant bunriKinshi | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. Type: Boolean or idNothingEnum enumerator. |
| characterAlignment | variant characterAlignment | The alignment of small characters to the largest character in the line. Type: idCharacterAlignment enumerator or idNothingEnum enumerator. |
| characterRotation | variant characterRotation | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. Type: Double or idNothingEnum enumerator. |
| cjkGridTracking | variant cjkGridTracking | If true, uses grid tracking to track non-Roman characters in CJK grids. Type: Boolean or idNothingEnum enumerator. |
| glyphForm | variant glyphForm | The glyph variant to substitute for standard glyphs. Type: idAlternateGlyphForms enumerator or idNothingEnum enumerator. |
| gridAlignment | variant gridAlignment | The alignment to the frame grid or baseline grid. Type: idGridAlignment enumerator or idNothingEnum enumerator. |
| gridGyoudori | variant gridGyoudori | The manual gyoudori setting. Type: Integer or idNothingEnum enumerator. |
| jidori | variant jidori | The number of grid squares in which to arrange the text. . Type: Integer or idNothingEnum enumerator. |
| keepWithPrevious | variant keepWithPrevious | If the first line in the paragraph should be kept with the last line of previous paragraph. Type: Boolean or idNothingEnum enumerator. |
| kntenAlignment | variant kntenAlignment | The alignment of knten characters relative to the parent characters. . Type: idKntenAlignment enumerator or idNothingEnum enumerator. |
| kntenCharacterSet | variant kntenCharacterSet | The character set used for the custom knten character. Note: Valid only when knten kind is custom. . Type: idKntenCharacterSet enumerator or idNothingEnum enumerator. |
| kntenCustomCharacter | variant kntenCustomCharacter | The character used for knten. Note: Valid only when knten kind is custom. Type: String or idNothingEnum enumerator. |
| kntenFillColor | variant kntenFillColor | The swatch (color, gradient, tint, or mixed ink) applied to the fill of knten characters. Type: Swatch, String or idNothingEnum enumerator. |
| kntenFont | variant kntenFont | The font to use for knten characters. Type: Font, String or idNothingEnum enumerator. |
| kntenFontSize | variant kntenFontSize | The size (in points) of knten characters. Type: Double or idNothingEnum enumerator. |
| kntenFontStyle | variant kntenFontStyle | The font style of knten characters. Type: String or idNothingEnum enumerator. |
| kntenKind | variant kntenKind | The style of knten characters. Type: idKntenCharacter enumerator or idNothingEnum enumerator. |

| FINDTEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--------------------------------|---|
| kntenOverprintFill | variant kntenOverprintFill | The method of overprinting the knten fill. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator. |
| kntenOverprintStroke | variant kntenOverprintStroke | The method of overprinting the knten stroke. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator. |
| kntenPlacement | variant kntenPlacement | The distance between knten characters and their parent characters. Type: Double or idNothingEnum enumerator. |
| kntenPosition | variant kntenPosition | The knten position relative to the parent character. Type: idRubyKetenPosition enumerator or idNothingEnum enumerator. |
| kntenStrokeColor | variant kntenStrokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of knten characters. Type: Swatch, String or idNothingEnum enumerator. |
| kntenStrokeTint | variant kntenStrokeTint | The stroke tint (as a percentage) of knten characters. (Range: 0 to 100). Type: Double or idNothingEnum enumerator. |
| kntenTint | variant kntenTint | The fill tint (as a percentage) of knten characters. (Range: 0 to 100). Type: Double or idNothingEnum enumerator. |
| kntenWeight | variant kntenWeight | The stroke weight (in points) of knten characters. Type: Double or idNothingEnum enumerator. |
| kntenXScale | variant kntenXScale | The horizontal size of knten characters as a percent of the original size. Type: Double or idNothingEnum enumerator. |
| kntenYScale | variant kntenYScale | The vertical size of knten characters as a percent of the original size. . Type: Double or idNothingEnum enumerator. |
| kinsokuHangType | variant kinsokuHangType | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. Type: idKinsokuHangTypes enumerator or idNothingEnum enumerator. |
| kinsokuSet | variant kinsokuSet | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator, String or idNothingEnum enumerator. |
| kinsokuType | variant kinsokuType | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. Type: idKinsokuType enumerator or idNothingEnum enumerator. |
| leadingAki | variant leadingAki | The amount of space before each character. Type: Double or idNothingEnum enumerator. |
| leadingModel | variant leadingModel | The point from which leading is measured from line to line. Type: idLeadingModel enumerator or idNothingEnum enumerator. |
| mojikumi | variant mojikumi | The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String, idMojikumiTableDefaults enumerator or idNothingEnum enumerator. |
| otfHVKana | variant otfHVKana | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. . Type: Boolean or idNothingEnum enumerator. |
| otfProportionalMetrics | variant otfProportionalMetrics | If true, kerns according to proportional CJK metrics in OpenType fonts. Type: Boolean or idNothingEnum enumerator. |
| otfRomanItalics | variant otfRomanItalics | If true, applies italics to half-width alphanumerics. . Type: Boolean or idNothingEnum enumerator. |
| paragraphGyoudori | variant paragraphGyoudori | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. Type: Boolean or idNothingEnum enumerator. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |

| FINDTEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| <code>rensuji</code> | <code>variant rensuji</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rotateSingleByteCharacters</code> | <code>variant rotateSingleByteCharacters</code> | If true, rotates Roman characters in vertical text. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAlignment</code> | <code>variant rubyAlignment</code> | The ruby alignment. Type: <code>idRubyAlignments</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoAlign</code> | <code>variant rubyAutoAlign</code> | If true, auto aligns ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoScaling</code> | <code>variant rubyAutoScaling</code> | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see <code>ruby parent scaling percent</code> . . Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyAutoScale</code> | <code>variant rubyAutoTcyAutoScale</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyDigits</code> | <code>variant rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. Type: Integer or <code>idNothingEnum</code> enumerator. |
| <code>rubyAutoTcyIncludeRoman</code> | <code>variant rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyFill</code> | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyFont</code> | <code>variant rubyFont</code> | The font applied to ruby characters. Type: Font, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyFontSize</code> | <code>variant rubyFontSize</code> | The size (in points) of ruby characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: String or <code>idNothingEnum</code> enumerator. |
| <code>rubyOpenTypePro</code> | <code>variant rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverhang</code> | <code>variant rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see <code>ruby parent overhang amount</code> . . Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverprintFill</code> | <code>variant rubyOverprintFill</code> | The method of overprinting the ruby fill. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyOverprintStroke</code> | <code>variant rubyOverprintStroke</code> | The method of overprinting the ruby stroke. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentOverhangAmount</code> | <code>variant rubyParentOverhangAmount</code> | The amount by which ruby characters can overhang the parent text. Type: <code>idRubyOverhang</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentScalingPercent</code> | <code>variant rubyParentScalingPercent</code> | The amount (as a percentage) to scale the parent text size to determine the ruby text size. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>rubyParentSpacing</code> | <code>variant rubyParentSpacing</code> | The ruby spacing relative to the parent text. . Type: <code>idRubyParentSpacing</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyPosition</code> | <code>variant rubyPosition</code> | The position of ruby characters relative to the parent text. Type: <code>idRubyKertenPosition</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| <code>rubyStrokeTint</code> | <code>variant rubyStrokeTint</code> | The stroke tint (as a percentage) of ruby characters. Type: Double or <code>idNothingEnum</code> enumerator. |

| FINDTEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---|---|
| rubyTint | <code>variant rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). Type: Double or idNothingEnum enumerator.</i> |
| rubyType | <code>variant rubyType</code> | <i>The ruby type. Type: idRubyTypes enumerator or idNothingEnum enumerator.</i> |
| rubyWeight | <code>variant rubyWeight</code> | <i>The stroke weight (in points) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyXOffset | <code>variant rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyXScale | <code>variant rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| rubyYOffset | <code>variant rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyYScale | <code>variant rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| scaleAffectsLineHeight | <code>variant scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled. . Type: Boolean or idNothingEnum enumerator.</i> |
| shataiAdjustRotation | <code>variant shataiAdjustRotation</code> | <i>If true, applies shatai rotation. Type: Boolean or idNothingEnum enumerator.</i> |
| shataiAdjustTsume | <code>variant shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume. Type: Boolean or idNothingEnum enumerator.</i> |
| shataiDegreeAngle | <code>variant shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees). Type: Double or idNothingEnum enumerator.</i> |
| shataiMagnification | <code>variant shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply. Type: Double or idNothingEnum enumerator.</i> |
| spanColumnMinSpaceAfter | <code>variant spanColumnMinSpaceAfter</code> | <i>The minimum space after a span or a split column. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| spanColumnMinSpaceBefore | <code>variant spanColumnMinSpaceBefore</code> | <i>The minimum space before a span or a split column. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| spanColumnType | <code>variant spanColumnType</code> | <i>Whether a paragraph should be a single column, span columns or split columns. Type: idSpanColumnTypeOptions enumerator or idNothingEnum enumerator.</i> |
| spanSplitColumnCount | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40), idSpanColumnCountOptions enumerator or idNothingEnum enumerator.</i> |
| splitColumnInsideGutter | <code>variant splitColumnInsideGutter</code> | <i>The inside gutter if the paragraph splits columns. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| splitColumnOutsideGutter | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| tatechuyoko | <code>variant tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text. Type: Boolean or idNothingEnum enumerator.</i> |
| tatechuyokoXOffset | <code>variant tatechuyokoXOffset</code> | <i>The horizontal offset for horizontal characters in vertical text. Type: Double or idNothingEnum enumerator.</i> |
| tatechuyokoYOffset | <code>variant tatechuyokoYOffset</code> | <i>The vertical offset for horizontal characters in vertical text. Type: Double or idNothingEnum enumerator.</i> |
| trailingAki | <code>variant trailingAki</code> | <i>The amount of space after each character. Type: Double or idNothingEnum enumerator.</i> |

| FINDTEXTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| <code>treatIdeographicSpaceAsSpace</code> | <code>variant treatIdeographicSpaceAsSpace</code> | If true, ideographic spaces will not wrap to the next line like text characters. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>tsume</code> | <code>variant tsume</code> | The amount of horizontal character compression. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>warichu</code> | <code>variant warichu</code> | If true, turns on warichu. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>warichuAlignment</code> | <code>variant warichuAlignment</code> | The warichu alignment. Type: <code>idWarichuAlignment</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>warichuCharsAfterBreak</code> | <code>variant warichuCharsAfterBreak</code> | The minimum number of characters allowed after a line break. . Type: Integer or <code>idNothingEnum</code> enumerator. |
| <code>warichuCharsBeforeBreak</code> | <code>variant warichuCharsBeforeBreak</code> | The minimum number of characters allowed before a line break. Type: Integer or <code>idNothingEnum</code> enumerator. |
| <code>warichuLines</code> | <code>variant warichuLines</code> | The number of lines of warichu within a single normal line. Type: Integer or <code>idNothingEnum</code> enumerator. |
| <code>warichuLineSpacing</code> | <code>variant warichuLineSpacing</code> | The gap between lines of warichu characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>warichuSize</code> | <code>variant warichuSize</code> | The amount (as a percentage) to scale parent text size to determine warichu size. Type: Double or <code>idNothingEnum</code> enumerator. |

FrameFittingOption

Object available in ID 5, 6, 7

| FRAMEFITTINGOPTION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|------------------------------|--|
| <code>autoFit</code> | <code>boolean autoFit</code> | If true, the last saved fitting options will be applied to the contents of a frame when it is resized. |

ImportedPage

Object available in ID 5, 6, 7

| IMPORTEDPAGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|---|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: <code>idExportFormat</code> enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: <code>PDFExportPreset</code> . / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| <code>clearTransformations</code> | <code>void clearTransformations()</code> | Clears transformations from the ImportedPage. Transformations include rotation, scaling, flipping, fitting, and shearing. |

| IMPORTEDPAGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|---|
| convertShape | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | Converts the ImportedPage to a different shape. / Given: The ImportedPage's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0) . Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points). |
| select | <code>void select(#SelectionOptions ExistingSelection)</code> | Selects the object. / ExistingSelection: The selection status of the ImportedPage in relation to previously selected objects. |
| store | <code>{Asset} store(Library Using)</code> | Stores the object in the specified library. / Return value: The new library asset(s). / Using: The library in which to store the object. |
| allGraphics | <code>{Objects} *allGraphics</code> | Lists all graphics contained by the ImportedPage. |
| allPageItems | <code>{Objects} *allPageItems</code> | Lists all page items contained by the ImportedPage. |
| animationSettings | <code>{AnimationSetting} *animationSettings</code> | The page item animation settings. |
| bottomLeftCornerOption | <code>#CornerOptions bottomLeftCornerOption</code> | The shape to apply to the bottom left corner of rectangular shapes. |
| bottomLeftCornerRadius | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| bottomRightCornerOption | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| bottomRightCornerRadius | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| endCap | <code>#EndCap endCap</code> | The end shape of an open path. |
| endJoin | <code>#EndJoin endJoin</code> | The corner join applied to the ImportedPage. |
| fillColor | <code>variant fillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of the ImportedPage. . Type: Swatch or String. |
| fillTint | <code>double fillTint</code> | The percent of tint to use in the ImportedPage's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| gapColor | <code>{Swatch} gapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type. |
| gapTint | <code>double gapTint</code> | The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| gradientFillAngle | <code>double gradientFillAngle</code> | The angle of a linear gradient applied to the fill of the ImportedPage. (Range: -180 to 180). |
| gradientFillLength | <code>variant gradientFillLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the ImportedPage. Type: Unit (Double or String). |
| gradientFillStart | <code>variant gradientFillStart</code> | The starting point (in page coordinates) of a gradient applied to the fill of the ImportedPage, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| gradientStrokeAngle | <code>double gradientStrokeAngle</code> | The angle of a linear gradient applied to the stroke of the ImportedPage. (Range: -180 to 180). |
| gradientStrokeLength | <code>variant gradientStrokeLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the ImportedPage. Type: Unit (Double or String). |
| gradientStrokeStart | <code>variant gradientStrokeStart</code> | The starting point (in page coordinates) of a gradient applied to the stroke of the ImportedPage, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |

| IMPORTEDPAGE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|--|---|
| itemLayer | {Layer} itemLayer | The layer that the ImportedPage is on. |
| leftLineEnd | #ArrowHead leftLineEnd | The arrowhead applied to the start of the path. |
| locked | boolean locked | If true, the ImportedPage is locked. |
| miterLimit | double miterLimit | The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500). |
| name | string name | The name of the ImportedPage. |
| overprintFill | boolean overprintFill | If true, the ImportedPage's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors. |
| overprintGap | boolean overprintGap | If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors. |
| overprintStroke | boolean overprintStroke | If true, the ImportedPage's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors. |
| parentPage | {Page} *parentPage | The page on which this page item appears. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |
| rightLineEnd | #ArrowHead rightLineEnd | The arrowhead applied to the end of the path. |
| strokeAlignment | #StrokeAlignment strokeAlignment | The stroke alignment applied to the ImportedPage. |
| strokeColor | variant strokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the ImportedPage. Type: Swatch or String. |
| strokeCornerAdjustment | #StrokeCornerAdjustment strokeCornerAdjustment | The corner adjustment applied to the ImportedPage. |
| strokeDashAndGap | variant strokeDashAndGap | The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings). |
| strokeTint | double strokeTint | The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| strokeType | variant strokeType | The name of the stroke style to apply. Type: StrokeStyle or String. |
| strokeWeight | variant strokeWeight | The weight (in points) to apply to the ImportedPage's stroke. Type: Unit (Double or String). |
| timingSettings | {TimingSetting} *timingSettings | The object timing settings. |
| topLeftCornerOption | #CornerOptions topLeftCornerOption | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. |
| topLeftCornerRadius | variant topLeftCornerRadius | The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String). |
| topRightCornerOption | #CornerOptions topRightCornerOption | The shape to apply to the top right corner of rectangular shapes. |
| topRightCornerRadius | variant topRightCornerRadius | The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String). |
| visible | boolean visible | If true, the ImportedPage is visible. |

PageItemDefault

Object available in ID 5, 6, 7

| PAGEITEMDEFAULT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|---|--|
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |

StrokeTransparencySetting

Object available in ID 5, 6, 7

| STROKETRANSPARENCYSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|---|---|
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |

TextVariable

Object available in ID 5, 6, 7

| TEXTVARIABLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|---|---|
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |

TransparencySetting

Object available in ID 5, 6, 7

| TRANSPARENCYSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---|---|
| <code>preferences</code> | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |

XMLRuleProcessor

Object available in ID 5, 6, 7

| XMLRULEPROCESSOR CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--------------------------|--|
| <code>name</code> | <code>string name</code> | The name of the XMLRuleProcessor; this is an alias to the XMLRuleProcessor's label property. |

XmlStory

Object available in ID 5, 6, 7

| XMLSTORY CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|--|
| <code>asynchronousExportFile</code> | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| <code>changeTransliterate</code> | <code>{Objects} changeTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value and replaces the text with the change character type value. / Return value: The replaced text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| <code>findTransliterate</code> | <code>{Objects} findTransliterate(variant ReverseOrder)</code> | Finds text that matches the find character type value. / Return value: The found text. / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| <code>allowArbitraryHyphenation</code> | <code>boolean allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. |
| <code>appliedNamedGrid</code> | <code>{NamedGrid} appliedNamedGrid</code> | The named grid in use. |
| <code>autoTcy</code> | <code>const autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. |
| <code>autoTcyIncludeRoman</code> | <code>boolean autoTcyIncludeRoman</code> | If true, auto tcy includes Roman characters. |
| <code>bunriKinshi</code> | <code>boolean bunriKinshi</code> | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. |
| <code>buttons</code> | <code>{Buttons} *buttons</code> | A collection of buttons. |
| <code>changes</code> | <code>{Changes} *changes</code> | A collection of changes. |
| <code>characterAlignment</code> | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| <code>characterRotation</code> | <code>double characterRotation</code> | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. |
| <code>cjkGridTracking</code> | <code>boolean cjkGridTracking</code> | If true, uses grid tracking to track non-Roman characters in CJK grids. |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| <code>formFields</code> | <code>{FormFields} *formFields</code> | A collection of form fields. |
| <code>glyphForm</code> | <code>#AlternateGlyphForms glyphForm</code> | The glyph variant to substitute for standard glyphs. |
| <code>graphicLines</code> | <code>{GraphicLines} *graphicLines</code> | A collection of graphic lines. |

| XML/STORY CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|--|
| <code>gridAlignment</code> | <code>#GridAlignment gridAlignment</code> | <i>The alignment to the frame grid or baseline grid.</i> |
| <code>gridData</code> | <code>{GridDataInformation} *gridData</code> | <i>Default grid properties. Note: Applies to named, layout, and frame (story) grids.</i> |
| <code>gridGyoudori</code> | <code>const gridGyoudori</code> | <i>The manual gyoudori setting.</i> |
| <code>groups</code> | <code>{Groups} *groups</code> | <i>A collection of groups.</i> |
| <code>hiddenTexts</code> | <code>{HiddenTexts} *hiddenTexts</code> | <i>A collection of hidden text objects.</i> |
| <code>jidori</code> | <code>const jidori</code> | <i>The number of grid squares in which to arrange the text.</i> |
| <code>keepWithPrevious</code> | <code>boolean keepWithPrevious</code> | <i>If the first line in the paragraph should be kept with the last line of previous paragraph.</i> |
| <code>kentenAlignment</code> | <code>#KentenAlignment kentenAlignment</code> | <i>The alignment of kenten characters relative to the parent characters.</i> |
| <code>kentenCharacterSet</code> | <code>#KentenCharacterSet kentenCharacterSet</code> | <i>The character set used for the custom kenten character. Note: Valid only when kenten kind is custom.</i> |
| <code>kentenCustomCharacter</code> | <code>string kentenCustomCharacter</code> | <i>The character used for kenten. Note: Valid only when kenten kind is custom.</i> |
| <code>kentenFillColor</code> | <code>variant kentenFillColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch or String.</i> |
| <code>kentenFont</code> | <code>variant kentenFont</code> | <i>The font to use for kenten characters. Type: Font or String.</i> |
| <code>kentenFontSize</code> | <code>double kentenFontSize</code> | <i>The size (in points) of kenten characters.</i> |
| <code>kentenFontStyle</code> | <code>variant kentenFontStyle</code> | <i>The font style of kenten characters. Type: String or idNothingEnum enumerator.</i> |
| <code>kentenKind</code> | <code>#KentenCharacter kentenKind</code> | <i>The style of kenten characters.</i> |
| <code>kentenOverprintFill</code> | <code>#AdornmentOverprint kentenOverprintFill</code> | <i>The method of overprinting the kenten fill.</i> |
| <code>kentenOverprintStroke</code> | <code>#AdornmentOverprint kentenOverprintStroke</code> | <i>The method of overprinting the kenten stroke.</i> |
| <code>kentenPlacement</code> | <code>double kentenPlacement</code> | <i>The distance between kenten characters and their parent characters.</i> |
| <code>kentenPosition</code> | <code>#RubyKetenPosition kentenPosition</code> | <i>The kenten position relative to the parent character.</i> |
| <code>kentenStrokeColor</code> | <code>variant kentenStrokeColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of kenten characters. Type: Swatch or String.</i> |
| <code>kentenStrokeTint</code> | <code>double kentenStrokeTint</code> | <i>The stroke tint (as a percentage) of kenten characters. (Range: 0 to 100).</i> |
| <code>kentenTint</code> | <code>double kentenTint</code> | <i>The fill tint (as a percentage) of kenten characters. (Range: 0 to 100).</i> |
| <code>kentenWeight</code> | <code>double kentenWeight</code> | <i>The stroke weight (in points) of kenten characters.</i> |
| <code>kentenXScale</code> | <code>double kentenXScale</code> | <i>The horizontal size of kenten characters as a percent of the original size.</i> |
| <code>kentenYScale</code> | <code>double kentenYScale</code> | <i>The vertical size of kenten characters as a percent of the original size.</i> |
| <code>kinsokuHangType</code> | <code>#KinsokuHangTypes kinsokuHangType</code> | <i>The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect.</i> |
| <code>kinsokuSet</code> | <code>variant kinsokuSet</code> | <i>The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator or String.</i> |
| <code>kinsokuType</code> | <code>#KinsokuType kinsokuType</code> | <i>The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined.</i> |
| <code>leadingAki</code> | <code>double leadingAki</code> | <i>The amount of space before each character.</i> |
| <code>leadingModel</code> | <code>#LeadingModel leadingModel</code> | <i>The point from which leading is measured from line to line.</i> |

| XMLSTORY CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|---|--|
| mojikumi | <code>variant mojikumi</code> | The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String or idMojikumiTableDefaults enumerator. |
| multiStateObjects | <code>{MultiStateObjects} *multiStateObjects</code> | A collection of multi-state objects. |
| name | <code>string name</code> | The name of the XmlStory; this is an alias to the XmlStory's label property. |
| nestedGrepStyles | <code>{NestedGrepStyles} *nestedGrepStyles</code> | A collection of nested GREP styles. |
| nestedLineStyles | <code>{NestedLineStyles} *nestedLineStyles</code> | A collection of nested line styles. |
| nestedStyles | <code>{NestedStyles} *nestedStyles</code> | A collection of nested styles. |
| otfHVKana | <code>boolean otfHVKana</code> | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. |
| otfProportionalMetrics | <code>boolean otfProportionalMetrics</code> | If true, kerns according to proportional CJK metrics in OpenType fonts. |
| otfRomanItalics | <code>boolean otfRomanItalics</code> | If true, applies italics to half-width alphanumerics. |
| ovals | <code>{Ovals} *ovals</code> | A collection of ellipses. |
| paragraphGyoudori | <code>boolean paragraphGyoudori</code> | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. |
| polygons | <code>{Polygons} *polygons</code> | A collection of polygons. |
| preferences | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| rectangles | <code>{Rectangles} *rectangles</code> | A collection of rectangles. |
| rensuumi | <code>boolean rensuumi</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. |
| rotateSingleByteCharacters | <code>boolean rotateSingleByteCharacters</code> | If true, rotates Roman characters in vertical text. |
| rubyAlignment | <code>#RubyAlignments rubyAlignment</code> | The ruby alignment. |
| rubyAutoAlign | <code>boolean rubyAutoAlign</code> | If true, auto aligns ruby. |
| rubyAutoScaling | <code>boolean rubyAutoScaling</code> | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. |
| rubyAutoTcyAutoSize | <code>boolean rubyAutoTcyAutoSize</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. |
| rubyAutoTcyDigits | <code>const rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. |
| rubyAutoTcyIncludeRoman | <code>boolean rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. |
| rubyFill | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch or String. |
| rubyFlag | <code>boolean rubyFlag</code> | If true, ruby is on. |
| rubyFont | <code>variant rubyFont</code> | The font applied to ruby characters. Type: Font or String. |
| rubyFontSize | <code>double rubyFontSize</code> | The size (in points) of ruby characters. |
| rubyFontStyle | <code>variant rubyFontStyle</code> | The font style of ruby characters. Type: String or idNothingEnum enumerator. |
| rubyOpenTypePro | <code>boolean rubyOpenTypePro</code> | If true, uses OpenType Pro fonts for ruby. |
| rubyOverhang | <code>boolean rubyOverhang</code> | If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. |
| rubyOverprintFill | <code>#AdornmentOverprint rubyOverprintFill</code> | The method of overprinting the ruby fill. |

| XMLSTORY CSS MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|--|---|
| <code>rubyOverprintStroke</code> | <code>#AdornmentOverprint rubyOverprintStroke</code> | <i>The method of overprinting the ruby stroke.</i> |
| <code>rubyParentOverhangAmount</code> | <code>#RubyOverhang rubyParentOverhangAmount</code> | <i>The amount by which ruby characters can overhang the parent text.</i> |
| <code>rubyParentScalingPercent</code> | <code>double rubyParentScalingPercent</code> | <i>The amount (as a percentage) to scale the parent text size to determine the ruby text size.</i> |
| <code>rubyParentSpacing</code> | <code>#RubyParentSpacing rubyParentSpacing</code> | <i>The ruby spacing relative to the parent text.</i> |
| <code>rubyPosition</code> | <code>#RubyKetenPosition rubyPosition</code> | <i>The position of ruby characters relative to the parent text.</i> |
| <code>rubyString</code> | <code>string rubyString</code> | <i>The ruby string contents.</i> |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch or String.</i> |
| <code>rubyStrokeTint</code> | <code>double rubyStrokeTint</code> | <i>The stroke tint (as a percentage) of ruby characters.</i> |
| <code>rubyTint</code> | <code>double rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100).</i> |
| <code>rubyType</code> | <code>#RubyTypes rubyType</code> | <i>The ruby type.</i> |
| <code>rubyWeight</code> | <code>double rubyWeight</code> | <i>The stroke weight (in points) of ruby characters.</i> |
| <code>rubyXOffset</code> | <code>double rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters.</i> |
| <code>rubyXScale</code> | <code>double rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size.</i> |
| <code>rubyYOffset</code> | <code>double rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters.</i> |
| <code>rubyYScale</code> | <code>double rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size.</i> |
| <code>scaleAffectsLineHeight</code> | <code>boolean scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled.</i> |
| <code>shataiAdjustRotation</code> | <code>boolean shataiAdjustRotation</code> | <i>If true, applies shatai rotation.</i> |
| <code>shataiAdjustTsume</code> | <code>boolean shataiAdjustTsume</code> | <i>If true, adjusts shatai tsume.</i> |
| <code>shataiDegreeAngle</code> | <code>double shataiDegreeAngle</code> | <i>The shatai lens angle (in degrees).</i> |
| <code>shataiMagnification</code> | <code>double shataiMagnification</code> | <i>The amount (as a percentage) of shatai obliquing to apply.</i> |
| <code>spanColumnMinSpaceAfter</code> | <code>variant spanColumnMinSpaceAfter</code> | <i>The minimum space after a span or a split column. Type: Unit (Double or String).</i> |
| <code>spanColumnMinSpaceBefore</code> | <code>variant spanColumnMinSpaceBefore</code> | <i>The minimum space before a span or a split column. Type: Unit (Double or String).</i> |
| <code>spanColumnType</code> | <code>#SpanColumnTypeOptions spanColumnType</code> | <i>Whether a paragraph should be a single column, span columns or split columns.</i> |
| <code>spanSplitColumnCount</code> | <code>variant spanSplitColumnCount</code> | <i>The number of columns a paragraph spans or the number of split columns. Type: Integer (1 - 40) or idSpanColumnCountOptions enumerator.</i> |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |
| <code>splitColumnInsideGutter</code> | <code>variant splitColumnInsideGutter</code> | <i>The inside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>splitColumnOutsideGutter</code> | <code>variant splitColumnOutsideGutter</code> | <i>The outside gutter if the paragraph splits columns. Type: Unit (Double or String).</i> |
| <code>tabStops</code> | <code>{TabStops} *tabStops</code> | <i>A collection of tab stops.</i> |
| <code>tatechuyoko</code> | <code>boolean tatechuyoko</code> | <i>If true, makes the character horizontal in vertical text.</i> |

| XMLSTORY CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--------------------------------------|---|
| tatechuyokoXOffset | double tatechuyokoXOffset | The horizontal offset for horizontal characters in vertical text. |
| tatechuyokoYOffset | double tatechuyokoYOffset | The vertical offset for horizontal characters in vertical text. |
| textFrames | {TextFrames} *textFrames | A collection of text frames. |
| trailingAki | double trailingAki | The amount of space after each character. |
| treatIdeographicSpaceAsSpace | boolean treatIdeographicSpaceAsSpace | If true, ideographic spaces will not wrap to the next line like text characters. |
| tsume | double tsume | The amount of horizontal character compression. |
| warichu | boolean warichu | If true, turns on warichu. |
| warichuAlignment | #WarichuAlignment warichuAlignment | The warichu alignment. |
| warichuCharsAfterBreak | const warichuCharsAfterBreak | The minimum number of characters allowed after a line break. |
| warichuCharsBeforeBreak | const warichuCharsBeforeBreak | The minimum number of characters allowed before a line break. |
| warichuLines | const warichuLines | The number of lines of warichu within a single normal line. |
| warichuLineSpacing | double warichuLineSpacing | The gap between lines of warichu characters. |
| warichuSize | double warichuSize | The amount (as a percentage) to scale parent text size to determine warichu size. |

FindChangeContentTransparencySetting

Object available in ID 6, 7

| FINDCHANGECONTENTTRANSPARENCYSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|----------------------------|--------------------------------------|
| preferences | {Preferences} *preferences | A collection of preferences objects. |

FindChangeFillTransparencySetting

Object available in ID 6, 7

| FINDCHANGEFILLTRANSPARENCYSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|----------------------------|--------------------------------------|
| preferences | {Preferences} *preferences | A collection of preferences objects. |

FindChangeStrokeTransparencySetting

Object available in ID 6, 7

| FINDCHANGESTROKETRANSPARENCYSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|----------------------------|--------------------------------------|
| preferences | {Preferences} *preferences | A collection of preferences objects. |

FindChangeTransparencySetting

Object available in ID 6, 7

| FINDCHANGETRANSPARECYSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|----------------------------|--------------------------------------|
| preferences | {Preferences} *preferences | A collection of preferences objects. |

HiddenText

Object available in ID 6, 7

| HIDDENTEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------|-------------|--|
| name | string name | The name of the HiddenText; this is an alias to the HiddenText's label property. |

SWFExportPreference

Object available in ID 6, 7

| SWFEXPORTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---|--|
| dynamicMediaHandling | #DynamicMediaHandlingOptions dynamicMediaHandling | The dynamic media handling options. |
| flattenTransparency | boolean flattenTransparency | Flatten transparency when exporting. |
| frameRate | integer frameRate | The frame rate in frames per second. Type: Long Integer (1 - 120). |
| pageTransitionOverride | #PageTransitionOverrideOptions pageTransitionOverride | The name of the page transition to use for all pages. |
| swfBackground | #SWFBackgroundOptions swfBackground | SWF background setting. |

XFLExportPreference

Object available in ID 6, 7

| XFLEXPORTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---|---|
| dynamicMediaHandling | #DynamicMediaHandlingOptions dynamicMediaHandling | The dynamic media handling options. |
| insertHyphens | boolean insertHyphens | If true, discretionary hyphenation points are inserted when TLF text is used. |

AnimationBehavior

Object available in ID 7

| ANIMATIONBEHAVIOR CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|---------------|--------------------------------|
| delete | void delete() | Deletes the AnimationBehavior. |

| ANIMATIONBEHAVIOR CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|---|--|
| extractLabel | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| insertLabel | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| animatedPageItem | <code>{PageItem} animatedPageItem</code> | The animation page item. |
| autoReverseOnRollOff | <code>boolean autoReverseOnRollOff</code> | If true, will automatically play the animation in reverse on roll off of the rollover event. |
| behaviorEvent | <code>#BehaviorEvents behaviorEvent</code> | The event which triggers the behavior. |
| enableBehavior | <code>boolean enableBehavior</code> | If true, the behavior is enabled. |
| id | <code>integer •id</code> | The unique ID of the AnimationBehavior. |
| index | <code>integer •index</code> | The index of the AnimationBehavior within its containing object. |
| label | <code>string label</code> | A property that can be set to any string. |
| name | <code>string •name</code> | The name of the AnimationBehavior. |
| operation | <code>#AnimationPlayOperations operation</code> | The playback mode. |
| parent | <code>{Button} •parent</code> | The parent of the AnimationBehavior (a Button). |

AnimationBehaviors

Object available in ID 7

| ANIMATIONBEHAVIORS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|--|
| add | <code>{AnimationBehavior} add()</code> | Creates a new AnimationBehavior. / Return value: The new AnimationBehavior. |
| anyItem | <code>IDispatch anyItem()</code> | Returns any object in the collection. / Return value: The object. |
| firstItem | <code>IDispatch firstItem()</code> | Returns the first object in the collection. / Return value: The first object. |
| itemByID | <code>IDispatch itemByID(integer Id)</code> | Returns the object with the specified ID. / Return value: The object. / ID: The ID. |
| itemByRange | <code>{Objects} itemByRange(variant From, variant To)</code> | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| lastItem | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |
| middleItem | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| nextItem | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| previousItem | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| count | <code>integer •count</code> | Displays the number of elements in the AnimationBehavior. |

| ANIMATIONBEHAVIORS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|--|
| item | {AnimationBehavior} *item(<i>variant</i> Index) | Returns the object with the specified index or name. Type: Long Integer or String. |

AnimationSetting

Object available in ID 7

| ANIMATIONSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|--|
| save | {MotionPreset} save(<i>string</i> Name) | Save this motion preset as custom preset. / Return value: The new custom motion preset. / Name: The name for the new motion preset. |
| saveACopy | void saveACopy(<i>string</i> To) | Save a copy of this motion preset to a InDesign motion preset file. / To: The Flash motion preset file to export to. |
| designOption | #DesignOptions designOption | The animation design option. |
| duration | double duration | The duration in second for this animation. Type: Double (0.125 - 60). |
| easeType | #AnimationEaseOptions easeType | The ease type. |
| hasCustomSettings | boolean *hasCustomSettings | Determines if this animated object has custom settings. |
| hiddenAfter | boolean hiddenAfter | Determines if this object is hidden after its animation is played in an exported SWF file. |
| initiallyHidden | boolean initiallyHidden | Determines if this object is initially hidden when displayed in an exported SWF file. |
| motionPath | <i>variant</i> motionPath | The list of motion path points and key frames for this animation. Type: Array of Ordered array containing KeyFrame:Long Integer, PathPoint:Ordered array containing Anchor:Array of 2 Doubles, LeftDirection:Array of 2 Doubles, RightDirection:Array of 2 Doubles. |
| motionPathPoints | <i>variant</i> motionPathPoints | The list of motion path points for this animation. Type: Ordered array containing PathPointArray:Array of Ordered array containing Anchor:Array of 2 Doubles, LeftDirection:Array of 2 Doubles, RightDirection:Array of 2 Doubles, PathOpen:Boolean. |
| opacityArray | <i>variant</i> opacityArray | The list of opacity key frames for this animation. Type: Array of Ordered array containing KeyFrame:Long Integer, Value:Double. |
| parent | <i>variant</i> *parent | The parent of the AnimationSetting (a PageItem, FormField, MultiStateObject, Button, MediaItem, Sound, Movie, EPSText, SplineItem, Polygon, GraphicLine, Rectangle, Oval, Graphic, ImportedPage, PICT, WMF, PDF, EPS, Image, Group or TextFrame). Type: PageItem, FormField, MultiStateObject, Button, MediaItem, Sound, Movie, EPSText, SplineItem, Polygon, GraphicLine, Rectangle, Oval, Graphic, ImportedPage, PICT, WMF, PDF, EPS, Image, Group or TextFrame. |
| plays | integer plays | The number of times this animation plays. Type: Long Integer (1 - 100). |
| playsLoop | boolean playsLoop | Set to true if animation play loops. |
| preset | <i>variant</i> preset | The base animation preset applied. Type: MotionPreset, String or idNothingEnum enumerator. |
| rotationArray | <i>variant</i> rotationArray | The list of rotation key frames for this animation. Type: Array of Ordered array containing KeyFrame:Long Integer, Value:Double. |

| ANIMATIONSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|---------------------------------------|--|
| <code>scaleXArray</code> | <code>variant scaleXArray</code> | <i>The list of scale x key frames for this animation. Type: Array of Ordered array containing KeyFrame:Long Integer, Value:Double.</i> |
| <code>scaleYArray</code> | <code>variant scaleYArray</code> | <i>The list of scale y key frames for this animation. Type: Array of Ordered array containing KeyFrame:Long Integer, Value:Double.</i> |
| <code>transformOffsets</code> | <code>variant transformOffsets</code> | <i>The transform offset percentage from the target object bounding box's left-top corner. Type: Array of 2 Doubles.</i> |

ApplicationEvents

Object available in ID 7

| APPLICATIONEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>afterActivate</code> | <code>const afterActivate(afterActivate)</code> | <i>Dispatched after the Application becomes active. This event bubbles. This event is not cancelable.</i> |
| <code>afterClose</code> | <code>const afterClose(afterClose)</code> | <i>Dispatched after a Document is closed. This event bubbles. This event is not cancelable.</i> |
| <code>afterContextChanged</code> | <code>const afterContextChanged(afterContextChanged)</code> | <i>Dispatched after the active context changes. This event bubbles. This event is not cancelable.</i> |
| <code>afterQuit</code> | <code>const afterQuit(afterQuit)</code> | <i>Dispatched when the Application is quitting. Since the quit has been committed, it can not be canceled. This event bubbles. This event is not cancelable.</i> |
| <code>afterSelectionAttributeChanged</code> | <code>const afterSelectionAttributeChanged(afterSelectionAttributeChanged)</code> | <i>Dispatched after an attribute on the active selection changes. This event bubbles. This event is not cancelable.</i> |
| <code>afterSelectionChanged</code> | <code>const afterSelectionChanged(afterSelectionChanged)</code> | <i>Dispatched after the active selection changes. This event bubbles. This event is not cancelable.</i> |
| <code>beforeDeactivate</code> | <code>const beforeDeactivate(beforeDeactivate)</code> | <i>Dispatched before the Application becomes inactive. This event bubbles. This event is not cancelable.</i> |
| <code>beforeNew</code> | <code>const beforeNew(beforeNew)</code> | <i>Dispatched before a Document is created. This event bubbles. This event is cancelable.</i> |
| <code>beforeOpen</code> | <code>const beforeOpen(beforeOpen)</code> | <i>Dispatched before a Document is opened. This event bubbles. This event is cancelable.</i> |
| <code>beforeQuit</code> | <code>const beforeQuit(beforeQuit)</code> | <i>Dispatched before the Application is quit. Allows the quit to be canceled. This event bubbles. This event is cancelable.</i> |

BackgroundTask

Object available in ID 7

| BACKGROUNDTASK CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|---|--|
| <code>cancelTask</code> | <code>void cancelTask()</code> | <i>Cancels the specified background task.</i> |
| <code>queryProperty</code> | <code>variant queryProperty(string Name)</code> | <i>Queries for a particular property in the task metadata. / Return value: the property value or void if the property is undefined. Type: Any Type. / Name: The task property being queried.</i> |

| BACKGROUNDTASK CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--------------------------|--|
| waitForTask | #TaskState waitForTask() | Waits for the task to finish. / Return value: The status of the task. |
| alerts | variant *alerts | The alerts encountered while running this task object. Type: Array of Array of 2 idTaskAlertType enumerators or Strings. |
| documentName | string *documentName | The document name on which this task operates. |
| id | integer *id | The unique ID of the BackgroundTask. |
| index | integer *index | The index of the BackgroundTask within its containing object. |
| name | string *name | The name of the BackgroundTask. |
| parent | {Application} *parent | The parent of the BackgroundTask (a Application). |
| percentDone | integer *percentDone | Progress information for this task. |
| status | #TaskState *status | The current status of this task object. |

BackgroundTasks

Object available in ID 7

| BACKGROUNDTASKS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|---|--|
| anyItem | IDispatch anyItem() | Returns any object in the collection. / Return value: The object. |
| firstItem | IDispatch firstItem() | Returns the first object in the collection. / Return value: The first object. |
| itemByID | IDispatch itemByID(integer Id) | Returns the object with the specified ID. / Return value: The object. / ID: The ID. |
| itemByRange | {Objects} itemByRange(variant From, variant To) | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| lastItem | IDispatch lastItem() | Returns the last object in the collection. / Return value: The last object. |
| middleItem | IDispatch middleItem() | Returns the middle object in the collection. / Return value: The middle object. |
| nextItem | IDispatch nextItem(IDispatch Obj) | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| previousItem | IDispatch previousItem(IDispatch Obj) | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| count | integer *count | Displays the number of elements in the BackgroundTask. |
| item | {BackgroundTask} *item(variant Index) | Returns the object with the specified index or name. Type: Long Integer or String. |

ButtonEvents

Object available in ID 7

| BUTTONEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|--|--|
| afterPlace | <code>const afterPlace(afterPlace)</code> | Dispatched after a Button is placed. This event bubbles. This event is not cancelable. |
| beforePlace | <code>const beforePlace(beforePlace)</code> | Dispatched before a Button is placed. This event bubbles. This event is cancelable. |

CaptionMetadataVariablePreference

Object available in ID 7

| CAPTIONMETADATAVARIABLEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| metadataProviderName | <code>string metadataProviderName</code> | Name of the metadata provider. |
| parent | <code>{TextVariable} •parent</code> | The parent of the CaptionMetadataVariablePreference (a TextVariable). |
| textAfter | <code>string textAfter</code> | The text that follows the value of the variable. (Limit: 128 characters). |
| textBefore | <code>string textBefore</code> | The text that precedes the value of the variable. (Limit: 128 characters). |

ChangeTransliteratePreference

Object available in ID 7

| CHANGETRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| allowArbitraryHyphenation | <code>variant allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. Type: Boolean or idNothingEnum enumerator. |
| appliedCharacterStyle | <code>variant appliedCharacterStyle</code> | The character style to search for or change to. Type: String, CharacterStyle or idNothingEnum enumerator. |
| appliedFont | <code>variant appliedFont</code> | The font applied to the ChangeTransliteratePreference, specified as either a font object or the name of font family. Type: Font, String or idNothingEnum enumerator. |
| appliedLanguage | <code>variant appliedLanguage</code> | The language to search for or change to. Type: String, Language, LanguageWithVendors or idNothingEnum enumerator. |
| appliedNumberingList | <code>variant appliedNumberingList</code> | The list to be part of. Type: NumberingList, String or idNothingEnum enumerator. |
| appliedParagraphStyle | <code>variant appliedParagraphStyle</code> | The paragraph style to search for or change to. Type: String, ParagraphStyle or idNothingEnum enumerator. |
| autoLeading | <code>variant autoLeading</code> | The percent of the type size to use for auto leading. (Range: 0 to 500). Type: Double or idNothingEnum enumerator. |
| autoTcy | <code>variant autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. Type: Integer or idNothingEnum enumerator. |

| CHANGETRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|-------------------------------------|--|
| autoTcyIncludeRoman | variant autoTcyIncludeRoman | If true, auto tcy includes Roman characters. Type: Boolean or idNothingEnum enumerator. |
| balanceRaggedLines | variant balanceRaggedLines | If true or set to an enumeration value, balances ragged lines. Note: Not valid with a single-line text composer. Type: Boolean, idBalanceLinesStyle enumerator or idNothingEnum enumerator. |
| baselineShift | variant baselineShift | The baseline shift applied to the text. Type: Unit (Double or String) or idNothingEnum enumerator. |
| bulletChar | variant •bulletChar | Bullet character. Type: Bullet or idNothingEnum enumerator. |
| bulletsAlignment | variant bulletsAlignment | The alignment of the bullet character. Type: idListAlignment enumerator or idNothingEnum enumerator. |
| bulletsAndNumberingListType | variant bulletsAndNumberingListType | List type for bullets and numbering. Type: idListType enumerator or idNothingEnum enumerator. |
| bulletsCharacterStyle | variant bulletsCharacterStyle | The character style to be used for the text after string. Type: CharacterStyle, String or idNothingEnum enumerator. |
| bulletsTextAfter | variant bulletsTextAfter | The text after string expression for bullets. Type: String or idNothingEnum enumerator. |
| bunriKinshi | variant bunriKinshi | If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. Type: Boolean or idNothingEnum enumerator. |
| capitalization | variant capitalization | The capitalization scheme. Type: idCapitalization enumerator or idNothingEnum enumerator. |
| changeCharacterType | variant changeCharacterType | The character type to which to change the found character type. Type: idFindChangeTransliterateCharacterTypes enumerator or idNothingEnum enumerator. |
| characterAlignment | variant characterAlignment | The alignment of small characters to the largest character in the line. Type: idCharacterAlignment enumerator or idNothingEnum enumerator. |
| characterRotation | variant characterRotation | The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. Type: Double or idNothingEnum enumerator. |
| cjkGridTracking | variant cjkGridTracking | If true, uses grid tracking to track non-Roman characters in CJK grids. Type: Boolean or idNothingEnum enumerator. |
| composer | variant composer | The text composer to use to compose the text. Type: String or idNothingEnum enumerator. |
| dropCapCharacters | variant dropCapCharacters | The number of characters to drop cap. Type: Integer (0 - 150) or idNothingEnum enumerator. |
| dropcapDetail | variant dropcapDetail | Details about the drop cap based on the glyph outlines. 1 = left side bearing. 2 = descenders. 0x100,0x200,0x400 are used for Japanese frame grid. Type: Long Integer or idNothingEnum enumerator. |
| dropCapLines | variant dropCapLines | The number of lines to drop cap. Type: Integer (0 - 25) or idNothingEnum enumerator. |
| fillColor | variant fillColor | The swatch (color, gradient, tint, or mixed ink), applied as a fill color, to search for or change to. Type: String, Swatch or idNothingEnum enumerator. |
| fillTint | variant fillTint | The tint (as a percentage) of the fill color of the ChangeTransliteratePreference. (To specify a tint percentage, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). Type: Double or idNothingEnum enumerator. |

| CHANGETRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|---|
| <code>firstLineIndent</code> | <code>variant firstLineIndent</code> | <i>The amount to indent the first line. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>fontStyle</code> | <code>variant fontStyle</code> | <i>The name of the font style. Type: String or idNothingEnum enumerator.</i> |
| <code>glyphForm</code> | <code>variant glyphForm</code> | <i>The glyph variant to substitute for standard glyphs. Type: idAlternateGlyphForms enumerator or idNothingEnum enumerator.</i> |
| <code>gradientFillAngle</code> | <code>variant gradientFillAngle</code> | <i>The angle of a linear gradient applied to the fill of the text. (Range: -180 to 180). Type: Double or idNothingEnum enumerator.</i> |
| <code>gradientFillLength</code> | <code>variant gradientFillLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the text. Type: Double or idNothingEnum enumerator.</i> |
| <code>gradientFillStart</code> | <code>variant gradientFillStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the fill of the text, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings) or idNothingEnum enumerator.</i> |
| <code>gradientStrokeAngle</code> | <code>variant gradientStrokeAngle</code> | <i>The angle of a linear gradient applied to the stroke of the text. (Range: -180 to 180). Type: Double or idNothingEnum enumerator.</i> |
| <code>gradientStrokeLength</code> | <code>variant gradientStrokeLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the text. Type: Double or idNothingEnum enumerator.</i> |
| <code>gradientStrokeStart</code> | <code>variant gradientStrokeStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the stroke of the text, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings) or idNothingEnum enumerator.</i> |
| <code>gridAlignment</code> | <code>variant gridAlignment</code> | <i>The alignment to the frame grid or baseline grid. Type: idGridAlignment enumerator or idNothingEnum enumerator.</i> |
| <code>gridGyoudori</code> | <code>variant gridGyoudori</code> | <i>The manual gyoudori setting. Type: Integer or idNothingEnum enumerator.</i> |
| <code>horizontalScale</code> | <code>variant horizontalScale</code> | <i>The horizontal scaling applied to the ChangeTransliteratePreference. Type: Double or idNothingEnum enumerator.</i> |
| <code>hyphenateAcrossColumns</code> | <code>variant hyphenateAcrossColumns</code> | <i>If true, allows the last word in a text column to be hyphenated. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>hyphenateLastWord</code> | <code>variant hyphenateLastWord</code> | <i>If true, allows hyphenation in the last word in a paragraph. Note: Valid only when hyphenation is true. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>ignoreEdgeAlignment</code> | <code>variant ignoreEdgeAlignment</code> | <i>If true, ignores optical edge alignment for the paragraph. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>jidori</code> | <code>variant jidori</code> | <i>The number of grid squares in which to arrange the text. . Type: Integer or idNothingEnum enumerator.</i> |
| <code>justification</code> | <code>variant justification</code> | <i>The paragraph alignment. Type: idJustification enumerator or idNothingEnum enumerator.</i> |
| <code>keepAllLinesTogether</code> | <code>variant keepAllLinesTogether</code> | <i>If true, keeps all lines of the paragraph together. If false, allows paragraphs to break across pages or columns. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>keepFirstLines</code> | <code>variant keepFirstLines</code> | <i>The minimum number of lines to keep together in a paragraph before allowing a page break. Type: Integer (1 - 50) or idNothingEnum enumerator.</i> |
| <code>keepLastLines</code> | <code>variant keepLastLines</code> | <i>The minimum number of lines to keep together in a paragraph after a page break. Type: Integer (1 - 50) or idNothingEnum enumerator.</i> |
| <code>keepLinesTogether</code> | <code>variant keepLinesTogether</code> | <i>If true, keeps a specified number of lines together when the paragraph breaks across columns or text frames. Type: Boolean or idNothingEnum enumerator.</i> |

| CHANGETRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| <code>keepRuleAboveInFrame</code> | <code>variant keepRuleAboveInFrame</code> | If true, forces the rule above the paragraph to remain in the frame bounds. Note: Valid only when rule above is true. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>keepWithNext</code> | <code>variant keepWithNext</code> | The minimum number of lines to keep with the next paragraph. Type: Integer (0 - 5) or <code>idNothingEnum</code> enumerator. |
| <code>kntenAlignment</code> | <code>variant kntenAlignment</code> | The alignment of knten characters relative to the parent characters. . Type: <code>idKntenAlignment</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>kntenCharacterSet</code> | <code>variant kntenCharacterSet</code> | The character set used for the custom knten character. Note: Valid only when knten kind is custom. . Type: <code>idKntenCharacterSet</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>kntenCustomCharacter</code> | <code>variant kntenCustomCharacter</code> | The character used for knten. Note: Valid only when knten kind is custom. Type: String or <code>idNothingEnum</code> enumerator. |
| <code>kntenFillColor</code> | <code>variant kntenFillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of knten characters. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| <code>kntenFont</code> | <code>variant kntenFont</code> | The font to use for knten characters. Type: Font, String or <code>idNothingEnum</code> enumerator. |
| <code>kntenFontSize</code> | <code>variant kntenFontSize</code> | The size (in points) of knten characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>kntenFontStyle</code> | <code>variant kntenFontStyle</code> | The font style of knten characters. Type: String or <code>idNothingEnum</code> enumerator. |
| <code>kntenKind</code> | <code>variant kntenKind</code> | The style of knten characters. Type: <code>idKntenCharacter</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>kntenOverprintFill</code> | <code>variant kntenOverprintFill</code> | The method of overprinting the knten fill. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>kntenOverprintStroke</code> | <code>variant kntenOverprintStroke</code> | The method of overprinting the knten stroke. Type: <code>idAdornmentOverprint</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>kntenPlacement</code> | <code>variant kntenPlacement</code> | The distance between knten characters and their parent characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>kntenPosition</code> | <code>variant kntenPosition</code> | The knten position relative to the parent character. Type: <code>idRubyKntenPosition</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>kntenStrokeColor</code> | <code>variant kntenStrokeColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of knten characters. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| <code>kntenStrokeTint</code> | <code>variant kntenStrokeTint</code> | The stroke tint (as a percentage) of knten characters. (Range: 0 to 100). Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>kntenTint</code> | <code>variant kntenTint</code> | The fill tint (as a percentage) of knten characters. (Range: 0 to 100). Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>kntenWeight</code> | <code>variant kntenWeight</code> | The stroke weight (in points) of knten characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>kntenXScale</code> | <code>variant kntenXScale</code> | The horizontal size of knten characters as a percent of the original size. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>kntenYScale</code> | <code>variant kntenYScale</code> | The vertical size of knten characters as a percent of the original size. . Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>kerningMethod</code> | <code>variant kerningMethod</code> | The type of pair kerning. Type: String or <code>idNothingEnum</code> enumerator. |
| <code>kerningValue</code> | <code>variant kerningValue</code> | The amount of space to add or remove between characters, specified in thousands of an em. . Type: Double or <code>idNothingEnum</code> enumerator. |

| CHANGETRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| kinsokuHangType | <i>variant</i> <code>kinsokuHangType</code> | <i>The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. Type: idKinsokuHangTypes enumerator or idNothingEnum enumerator.</i> |
| kinsokuSet | <i>variant</i> <code>kinsokuSet</code> | <i>The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator, String or idNothingEnum enumerator.</i> |
| kinsokuType | <i>variant</i> <code>kinsokuType</code> | <i>The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. Type: idKinsokuType enumerator or idNothingEnum enumerator.</i> |
| lastLineIndent | <i>variant</i> <code>lastLineIndent</code> | <i>The amount to indent the last line in the paragraph. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| leading | <i>variant</i> <code>leading</code> | <i>The leading applied to the text. Type: Unit (Double or String), idLeading enumerator or idNothingEnum enumerator.</i> |
| leadingAki | <i>variant</i> <code>leadingAki</code> | <i>The amount of space before each character. Type: Double or idNothingEnum enumerator.</i> |
| leadingModel | <i>variant</i> <code>leadingModel</code> | <i>The point from which leading is measured from line to line. Type: idLeadingModel enumerator or idNothingEnum enumerator.</i> |
| leftIndent | <i>variant</i> <code>leftIndent</code> | <i>The width of the left indent. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| ligatures | <i>variant</i> <code>ligatures</code> | <i>If true, replaces specific character combinations (e.g., fl, fi) with ligature characters. Type: Boolean or idNothingEnum enumerator.</i> |
| markupTag | <i>variant</i> <code>markupTag</code> | <i>The XML tag applied to the element. Type: String, XMLTag or idNothingEnum enumerator.</i> |
| mojikumi | <i>variant</i> <code>mojikumi</code> | <i>The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String, idMojikumiTableDefaults enumerator or idNothingEnum enumerator.</i> |
| noBreak | <i>variant</i> <code>noBreak</code> | <i>If true, keeps the text on the same line. . Type: Boolean or idNothingEnum enumerator.</i> |
| numberingAlignment | <i>variant</i> <code>numberingAlignment</code> | <i>The alignment of the number. Type: idListAlignment enumerator or idNothingEnum enumerator.</i> |
| numberingApplyRestartPolicy | <i>variant</i> <code>numberingApplyRestartPolicy</code> | <i>If true, apply the numbering restart policy. Type: Boolean or idNothingEnum enumerator.</i> |
| numberingCharacterStyle | <i>variant</i> <code>numberingCharacterStyle</code> | <i>The character style to be used for the number string. Type: CharacterStyle, String or idNothingEnum enumerator.</i> |
| numberingContinue | <i>variant</i> <code>numberingContinue</code> | <i>Continue the numbering at this level. Type: Boolean or idNothingEnum enumerator.</i> |
| numberingExpression | <i>variant</i> <code>numberingExpression</code> | <i>The number string expression for numbering. Type: String or idNothingEnum enumerator.</i> |
| numberingFormat | <i>variant</i> <code>numberingFormat</code> | <i>Numbering format options. Type: idNumberingStyle enumerator, String or idNothingEnum enumerator.</i> |
| numberingLevel | <i>variant</i> <code>numberingLevel</code> | <i>The level of the paragraph. Type: Long Integer or idNothingEnum enumerator.</i> |
| numberingStartAt | <i>variant</i> <code>numberingStartAt</code> | <i>Determines starting number in a numbered list. Type: Long Integer or idNothingEnum enumerator.</i> |
| otfContextualAlternate | <i>variant</i> <code>otfContextualAlternate</code> | <i>If true, uses contextual alternate forms in OpenType fonts. Type: Boolean or idNothingEnum enumerator.</i> |

| CHANGETRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| otfDiscretionaryLigature | <code>variant otfDiscretionaryLigature</code> | If true, uses discretionary ligatures in OpenType fonts. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfFigureStyle | <code>variant otfFigureStyle</code> | The figure style in OpenType fonts. Type: <code>idOTFFigureStyle</code> enumerator or <code>idNothingEnum</code> enumerator. |
| otfFraction | <code>variant otfFraction</code> | If true, uses fractions in OpenType fonts. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfHistorical | <code>variant otfHistorical</code> | If true, use historical forms in OpenType fonts. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfHVKana | <code>variant otfHVKana</code> | If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. . Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfLocale | <code>variant otfLocale</code> | If true, uses localized forms in OpenType fonts. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfMark | <code>variant otfMark</code> | If true, uses mark positioning in OpenType fonts. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfOrdinal | <code>variant otfOrdinal</code> | If true, uses ordinals in OpenType fonts. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfProportionalMetrics | <code>variant otfProportionalMetrics</code> | If true, kerns according to proportional CJK metrics in OpenType fonts. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfRomanItalics | <code>variant otfRomanItalics</code> | If true, applies italics to half-width alphanumerics. . Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfSlashedZero | <code>variant otfSlashedZero</code> | If true, use a slashed zeroes in OpenType fonts. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfStylisticSets | <code>variant otfStylisticSets</code> | The stylistic sets to use in OpenType fonts. Type: Long Integer or <code>idNothingEnum</code> enumerator. |
| otfSwash | <code>variant otfSwash</code> | If true, uses swash forms in OpenType fonts. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| otfTitling | <code>variant otfTitling</code> | If true, uses titling forms in OpenType fonts. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| overprintFill | <code>variant overprintFill</code> | If true, the fill color of the characters will overprint. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| overprintStroke | <code>variant overprintStroke</code> | If true, the stroke of the characters will overprint. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| paragraphGyoudori | <code>variant paragraphGyoudori</code> | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| parent | <code>{Application} *parent</code> | The parent of the <code>ChangeTransliteratePreference</code> (a Application). |
| pointSize | <code>variant pointSize</code> | The text size. Type: Unit (Double or String) or <code>idNothingEnum</code> enumerator. |
| position | <code>variant position</code> | The text position relative to the baseline. Type: <code>idPosition</code> enumerator or <code>idNothingEnum</code> enumerator. |
| positionalForm | <code>variant positionalForm</code> | The OpenType positional form. Type: <code>idPositionalForms</code> enumerator or <code>idNothingEnum</code> enumerator. |
| preferences | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| rensuuji | <code>variant rensuuji</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. Type: Boolean or <code>idNothingEnum</code> enumerator. |

| CHANGETRANSLITEREPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| <code>rightIndent</code> | <code>variant rightIndent</code> | <i>The width of the right indent. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| <code>rotateSingleByteCharacters</code> | <code>variant rotateSingleByteCharacters</code> | <i>If true, rotates Roman characters in vertical text. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>rubyAlignment</code> | <code>variant rubyAlignment</code> | <i>The ruby alignment. Type: idRubyAlignments enumerator or idNothingEnum enumerator.</i> |
| <code>rubyAutoAlign</code> | <code>variant rubyAutoAlign</code> | <i>If true, auto aligns ruby. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>rubyAutoScaling</code> | <code>variant rubyAutoScaling</code> | <i>If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. . Type: Boolean or idNothingEnum enumerator.</i> |
| <code>rubyAutoTcyAutoScale</code> | <code>variant rubyAutoTcyAutoScale</code> | <i>If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>rubyAutoTcyDigits</code> | <code>variant rubyAutoTcyDigits</code> | <i>The number of digits included in auto tcy (tate-chuu-yoko) in ruby. Type: Integer or idNothingEnum enumerator.</i> |
| <code>rubyAutoTcyIncludeRoman</code> | <code>variant rubyAutoTcyIncludeRoman</code> | <i>If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>rubyFill</code> | <code>variant rubyFill</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch, String or idNothingEnum enumerator.</i> |
| <code>rubyFont</code> | <code>variant rubyFont</code> | <i>The font applied to ruby characters. Type: Font, String or idNothingEnum enumerator.</i> |
| <code>rubyFontSize</code> | <code>variant rubyFontSize</code> | <i>The size (in points) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| <code>rubyFontStyle</code> | <code>variant rubyFontStyle</code> | <i>The font style of ruby characters. Type: String or idNothingEnum enumerator.</i> |
| <code>rubyOpenTypePro</code> | <code>variant rubyOpenTypePro</code> | <i>If true, uses OpenType Pro fonts for ruby. Type: Boolean or idNothingEnum enumerator.</i> |
| <code>rubyOverhang</code> | <code>variant rubyOverhang</code> | <i>If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. . Type: Boolean or idNothingEnum enumerator.</i> |
| <code>rubyOverprintFill</code> | <code>variant rubyOverprintFill</code> | <i>The method of overprinting the ruby fill. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator.</i> |
| <code>rubyOverprintStroke</code> | <code>variant rubyOverprintStroke</code> | <i>The method of overprinting the ruby stroke. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator.</i> |
| <code>rubyParentOverhangAmount</code> | <code>variant rubyParentOverhangAmount</code> | <i>The amount by which ruby characters can overhang the parent text. Type: idRubyOverhang enumerator or idNothingEnum enumerator.</i> |
| <code>rubyParentScalingPercent</code> | <code>variant rubyParentScalingPercent</code> | <i>The amount (as a percentage) to scale the parent text size to determine the ruby text size. Type: Double or idNothingEnum enumerator.</i> |
| <code>rubyParentSpacing</code> | <code>variant rubyParentSpacing</code> | <i>The ruby spacing relative to the parent text. . Type: idRubyParentSpacing enumerator or idNothingEnum enumerator.</i> |
| <code>rubyPosition</code> | <code>variant rubyPosition</code> | <i>The position of ruby characters relative to the parent text. Type: idRubyKertenPosition enumerator or idNothingEnum enumerator.</i> |
| <code>rubyStroke</code> | <code>variant rubyStroke</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch, String or idNothingEnum enumerator.</i> |
| <code>rubyStrokeTint</code> | <code>variant rubyStrokeTint</code> | <i>The stroke tint (as a percentage) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |

| CHANGETRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| rubyTint | <code>variant rubyTint</code> | The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). Type: Double or idNothingEnum enumerator. |
| rubyType | <code>variant rubyType</code> | The ruby type. Type: idRubyTypes enumerator or idNothingEnum enumerator. |
| rubyWeight | <code>variant rubyWeight</code> | The stroke weight (in points) of ruby characters. Type: Double or idNothingEnum enumerator. |
| rubyXOffset | <code>variant rubyXOffset</code> | The amount of horizontal space between ruby and parent characters. Type: Double or idNothingEnum enumerator. |
| rubyXScale | <code>variant rubyXScale</code> | The horizontal size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator. |
| rubyYOffset | <code>variant rubyYOffset</code> | The amount of vertical space between ruby and parent characters. Type: Double or idNothingEnum enumerator. |
| rubyYScale | <code>variant rubyYScale</code> | The vertical size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator. |
| scaleAffectsLineHeight | <code>variant scaleAffectsLineHeight</code> | If true, the line changes size when characters are scaled. . Type: Boolean or idNothingEnum enumerator. |
| shataiAdjustRotation | <code>variant shataiAdjustRotation</code> | If true, applies shatai rotation. Type: Boolean or idNothingEnum enumerator. |
| shataiAdjustTsume | <code>variant shataiAdjustTsume</code> | If true, adjusts shatai tsume. Type: Boolean or idNothingEnum enumerator. |
| shataiDegreeAngle | <code>variant shataiDegreeAngle</code> | The shatai lens angle (in degrees). Type: Double or idNothingEnum enumerator. |
| shataiMagnification | <code>variant shataiMagnification</code> | The amount (as a percentage) of shatai obliquing to apply. Type: Double or idNothingEnum enumerator. |
| singleWordJustification | <code>variant singleWordJustification</code> | The alignment to use for lines that contain a single word. Type: idSingleWordJustification enumerator or idNothingEnum enumerator. |
| skew | <code>variant skew</code> | The skew angle of the ChangeTransliteratePreference. Type: Double or idNothingEnum enumerator. |
| spaceAfter | <code>variant spaceAfter</code> | The height of the paragraph space below. Type: Unit (Double or String) or idNothingEnum enumerator. |
| spaceBefore | <code>variant spaceBefore</code> | The height of the paragraph space above. Type: Unit (Double or String) or idNothingEnum enumerator. |
| startParagraph | <code>variant startParagraph</code> | The location at which to start the paragraph. Type: idStartParagraph enumerator or idNothingEnum enumerator. |
| strikeThroughColor | <code>variant strikeThroughColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the strikethrough stroke. Type: Swatch, String or idNothingEnum enumerator. |
| strikeThroughGapColor | <code>variant strikeThroughGapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of the strikethrough stroke. Type: Swatch, String or idNothingEnum enumerator. |
| strikeThroughGapOverprint | <code>variant strikeThroughGapOverprint</code> | If true, the gap color of the strikethrough stroke will overprint. Note: Valid when strike through type is not solid. Type: Boolean or idNothingEnum enumerator. |
| strikeThroughGapTint | <code>variant strikeThroughGapTint</code> | The tint (as a percentage) of the strikethrough stroke gap color. (Range: 0 to 100) Note: Valid when strike through type is not solid. Type: Double or idNothingEnum enumerator. |
| strikeThroughOffset | <code>variant strikeThroughOffset</code> | The amount by which to offset the strikethrough stroke from the text baseline. Type: Unit (Double or String) or idNothingEnum enumerator. |

| CHANGETRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|---|
| strikeThroughOverprint | <code>variant strikeThroughOverprint</code> | If true, the strikethrough stroke will overprint. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| strikeThroughTint | <code>variant strikeThroughTint</code> | The tint (as a percentage) of the strikethrough stroke. (Range: 0 to 100). Type: Double or <code>idNothingEnum</code> enumerator. |
| strikeThroughType | <code>variant strikeThroughType</code> | The stroke type of the strikethrough stroke. Type: <code>StrokeStyle</code> , String or <code>idNothingEnum</code> enumerator. |
| strikeThroughWeight | <code>variant strikeThroughWeight</code> | The stroke weight of the strikethrough stroke. Type: Unit (Double or String) or <code>idNothingEnum</code> enumerator. |
| strikeThru | <code>variant strikeThru</code> | If true, draws a strikethrough line through the text. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| strokeColor | <code>variant strokeColor</code> | The swatch (color, gradient, tint, or mixed ink), applied as a stroke color, to search for or change to. Type: String, Swatch or <code>idNothingEnum</code> enumerator. |
| strokeTint | <code>variant strokeTint</code> | The tint (as a percentage) of the stroke color of the <code>ChangeTransliteratePreference</code> . (To specify a tint percentage, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). Type: Double or <code>idNothingEnum</code> enumerator. |
| strokeWeight | <code>variant strokeWeight</code> | The stroke weight applied to the characters of the text. Type: Unit (Double or String) or <code>idNothingEnum</code> enumerator. |
| tatechuyoko | <code>variant tatechuyoko</code> | If true, makes the character horizontal in vertical text. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| tatechuyokoXOffset | <code>variant tatechuyokoXOffset</code> | The horizontal offset for horizontal characters in vertical text. Type: Double or <code>idNothingEnum</code> enumerator. |
| tatechuyokoYOffset | <code>variant tatechuyokoYOffset</code> | The vertical offset for horizontal characters in vertical text. Type: Double or <code>idNothingEnum</code> enumerator. |
| tracking | <code>variant tracking</code> | The amount by which to loosen or tighten a block of text, specified in thousands of an em. Type: Double or <code>idNothingEnum</code> enumerator. |
| trailingAki | <code>variant trailingAki</code> | The amount of space after each character. Type: Double or <code>idNothingEnum</code> enumerator. |
| treatIdeographicSpaceAsSpace | <code>variant treatIdeographicSpaceAsSpace</code> | If true, ideographic spaces will not wrap to the next line like text characters. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| tsume | <code>variant tsume</code> | The amount of horizontal character compression. Type: Double or <code>idNothingEnum</code> enumerator. |
| underline | <code>variant underline</code> | If true, underlines the text. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| underlineColor | <code>variant underlineColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the underline stroke. . Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| underlineGapColor | <code>variant underlineGapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of the underline stroke. Note: Valid when underline type is not solid. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| underlineGapOverprint | <code>variant underlineGapOverprint</code> | If true, the gap color of the underline stroke will overprint. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| underlineGapTint | <code>variant underlineGapTint</code> | The tint (as a percentage) of the gap color of the underline stroke. (Range: 0 to 100) Note: Valid when underline type is not solid. Type: Double or <code>idNothingEnum</code> enumerator. |
| underlineOffset | <code>variant underlineOffset</code> | The amount by which to offset the underline from the text baseline. Type: Unit (Double or String) or <code>idNothingEnum</code> enumerator. |

| CHANGETRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>underlineOverprint</code> | <code>variant underlineOverprint</code> | If true, the underline stroke color will overprint. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>underlineTint</code> | <code>variant underlineTint</code> | The underline stroke tint (as a percentage). (Range: 0 to 100). Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>underlineType</code> | <code>variant underlineType</code> | The stroke type of the underline stroke. Type: <code>StrokeStyle</code> , String or <code>idNothingEnum</code> enumerator. |
| <code>underlineWeight</code> | <code>variant underlineWeight</code> | The stroke weight of the underline stroke. Type: Unit (Double or String) or <code>idNothingEnum</code> enumerator. |
| <code>verticalScale</code> | <code>variant verticalScale</code> | The vertical scaling applied to the <code>ChangeTransliteratePreference</code> . Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>warichu</code> | <code>variant warichu</code> | If true, turns on warichu. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| <code>warichuAlignment</code> | <code>variant warichuAlignment</code> | The warichu alignment. Type: <code>idWarichuAlignment</code> enumerator or <code>idNothingEnum</code> enumerator. |
| <code>warichuCharsAfterBreak</code> | <code>variant warichuCharsAfterBreak</code> | The minimum number of characters allowed after a line break. . Type: Integer or <code>idNothingEnum</code> enumerator. |
| <code>warichuCharsBeforeBreak</code> | <code>variant warichuCharsBeforeBreak</code> | The minimum number of characters allowed before a line break. Type: Integer or <code>idNothingEnum</code> enumerator. |
| <code>warichuLines</code> | <code>variant warichuLines</code> | The number of lines of warichu within a single normal line. Type: Integer or <code>idNothingEnum</code> enumerator. |
| <code>warichuLineSpacing</code> | <code>variant warichuLineSpacing</code> | The gap between lines of warichu characters. Type: Double or <code>idNothingEnum</code> enumerator. |
| <code>warichuSize</code> | <code>variant warichuSize</code> | The amount (as a percentage) to scale parent text size to determine warichu size. Type: Double or <code>idNothingEnum</code> enumerator. |

CjkGridPreference

Object available in ID 7

| CJKGRIDPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|---|---|
| <code>colorEveryNthCell</code> | <code>const colorEveryNthCell</code> | Applies the grid color to every nth cell, where n is the value of this property. |
| <code>iCFMode</code> | <code>boolean iCFMode</code> | If true, uses ICF mode for grid cells. If false, uses virtual body mode. |
| <code>layoutGridColorIndex</code> | <code>variant layoutGridColorIndex</code> | The layout grid color, specified either as an array of three doubles, each in the range 0 to 255 and representing R, G, and B values, or as a UI color. Type: Array of 3 Doubles (0 - 255) or <code>idUIColors</code> enumerator. |
| <code>minimumScale</code> | <code>double minimumScale</code> | The view magnification (as a percentage) less than which grids do not appear. (Range: 5 to 4000). |
| <code>parent</code> | <code>variant *parent</code> | The parent of the <code>CjkGridPreference</code> (a Application or Document). Type: Application or Document. |
| <code>showAllFrameGrids</code> | <code>boolean showAllFrameGrids</code> | If true, displays the frame (story) grids. |
| <code>showAllLayoutGrids</code> | <code>boolean showAllLayoutGrids</code> | If true, displays the layout grids. |
| <code>showCharacterCount</code> | <code>boolean showCharacterCount</code> | If true, displays the character count for the frame. |

| CJKGRIDPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|---|
| <code>singleLineColorMode</code> | <code>boolean singleLineColorMode</code> | If true, applies the grid color from the the edge of the line. If false, applies the grid color from the corner of the frame. |
| <code>snapToLayoutGrid</code> | <code>boolean snapToLayoutGrid</code> | If true, objects snap to the layout grid. |
| <code>useCircularCells</code> | <code>boolean useCircularCells</code> | If true, cell shape is circular. If false, cell shape is rectangular. |

CompositeFont

Object available in ID 7

| COMPOSITEFONT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|---|--|
| <code>delete</code> | <code>void delete()</code> | Deletes the CompositeFont. |
| <code>extractLabel</code> | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| <code>insertLabel</code> | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| <code>compositeFontEntries</code> | <code>{CompositeFontEntries} *compositeFontEntries</code> | A collection of composite font entries. |
| <code>id</code> | <code>integer *id</code> | The unique ID of the CompositeFont. |
| <code>index</code> | <code>integer *index</code> | The index of the CompositeFont within its containing object. |
| <code>label</code> | <code>string label</code> | A property that can be set to any string. |
| <code>name</code> | <code>string name</code> | The name of the CompositeFont. |
| <code>parent</code> | <code>variant *parent</code> | The parent of the CompositeFont (a Application or Document). Type: Application or Document. |

CompositeFontEntries

Object available in ID 7

| COMPOSITEFONTENTRIES CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|--|
| <code>add</code> | <code>{CompositeFontEntry} add()</code> | Creates a new CompositeFontEntry. / Return value: The new CompositeFontEntry. |
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | Returns any object in the collection. / Return value: The object. |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | Returns the first object in the collection. / Return value: The first object. |
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | Returns the object with the specified ID. / Return value: The object. / ID: The ID. |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |

| COMPOSITEFONTENTRIES CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|--|
| middleItem | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| nextItem | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| previousItem | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| count | <code>integer *count</code> | Displays the number of elements in the CompositeFontEntry. |
| item | <code>{CompositeFontEntry} *item(variant Index)</code> | Returns the object with the specified index or name. Type: Long Integer or String. |

CompositeFontEntry

Object available in ID 7

| COMPOSITEFONTENTRY CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|---|---|
| delete | <code>void delete()</code> | Deletes the CompositeFontEntry. |
| extractLabel | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| insertLabel | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| appliedFont | <code>variant appliedFont</code> | The font applied to the CompositeFontEntry, specified as either a font object or the name of font family. Type: Font or String. |
| baselineShift | <code>double baselineShift</code> | The amount of baseline shift. |
| customCharacters | <code>string customCharacters</code> | The characters that the set affects. |
| fontStyle | <code>string fontStyle</code> | The name of the font style. |
| horizontalScale | <code>double horizontalScale</code> | The horizontal scaling applied to the CompositeFontEntry. |
| id | <code>integer *id</code> | The unique ID of the CompositeFontEntry. |
| index | <code>integer *index</code> | The index of the CompositeFontEntry within its containing object. |
| label | <code>string label</code> | A property that can be set to any string. |
| locked | <code>boolean *locked</code> | If true, the CompositeFontEntry is locked. |
| name | <code>string name</code> | The name of the CompositeFontEntry. |
| parent | <code>{CompositeFont} *parent</code> | The parent of the CompositeFontEntry (a CompositeFont). |
| relativeSize | <code>double relativeSize</code> | The size of the entry relative to the base entry. Note: The base entry cannot be modified. |
| scaleOption | <code>boolean scaleOption</code> | If true, scales characters from the center. |
| verticalScale | <code>double verticalScale</code> | The vertical scaling applied to the CompositeFontEntry. |

CompositeFonts

Object available in ID 7

| COMPOSITEFONTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|---|--|
| add | {CompositeFont} add() | Creates a new CompositeFont. / Return value: The new CompositeFont. |
| anyItem | IDispatch anyItem() | Returns any object in the collection. / Return value: The object. |
| firstItem | IDispatch firstItem() | Returns the first object in the collection. / Return value: The first object. |
| itemByID | IDispatch itemByID(integer Id) | Returns the object with the specified ID. / Return value: The object. / ID: The ID. |
| itemByRange | {Objects} itemByRange(variant From, variant To) | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| lastItem | IDispatch lastItem() | Returns the last object in the collection. / Return value: The last object. |
| middleItem | IDispatch middleItem() | Returns the middle object in the collection. / Return value: The middle object. |
| nextItem | IDispatch nextItem(IDispatch Obj) | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| previousItem | IDispatch previousItem(IDispatch Obj) | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| count | integer *count | Displays the number of elements in the CompositeFont. |
| item | {CompositeFont} *item(variant Index) | Returns the object with the specified index or name. Type: Long Integer or String. |

DocumentEventEvents

Object available in ID 7

| DOCUMENTEVENTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---------------------------------------|---|
| afterClose | const afterClose(afterClose) | Dispatched after a Document is closed. This event bubbles. This event is not cancelable. |
| afterNew | const afterNew(afterNew) | Dispatched after a Document is created. This event bubbles. This event is not cancelable. |
| afterOpen | const afterOpen(afterOpen) | Dispatched after a Document is opened. This event bubbles. This event is not cancelable. |
| afterRevert | const afterRevert(afterRevert) | Dispatched after a Document is reverted. This event bubbles. This event is not cancelable. |
| afterSave | const afterSave(afterSave) | Dispatched after a Document is saved. This event bubbles. This event is not cancelable. |
| afterSaveACopy | const afterSaveACopy(afterSaveACopy) | Dispatched after a copy of a Document is saved. This event bubbles. This event is not cancelable. |

| DOCUMENTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|--|
| afterSaveAs | <code>const afterSaveAs(afterSaveAs)</code> | Dispatched after a Document is saved under a new name. This event bubbles. This event is not cancelable. |
| beforeClose | <code>const beforeClose(beforeClose)</code> | Dispatched before a Document is closed. This event bubbles. This event is not cancelable. |
| beforeNew | <code>const beforeNew(beforeNew)</code> | Dispatched before a Document is created. This event bubbles. This event is cancelable. |
| beforeOpen | <code>const beforeOpen(beforeOpen)</code> | Dispatched before a Document is opened. This event bubbles. This event is cancelable. |
| beforeRevert | <code>const beforeRevert(beforeRevert)</code> | Dispatched before a Document is reverted. This event bubbles. This event is cancelable. |
| beforeSave | <code>const beforeSave(beforeSave)</code> | Dispatched before a Document is saved. This event bubbles. This event is cancelable. |
| beforeSaveACopy | <code>const beforeSaveACopy(beforeSaveACopy)</code> | Dispatched before a copy of a Document is saved. This event bubbles. This event is cancelable. |
| beforeSaveAs | <code>const beforeSaveAs(beforeSaveAs)</code> | Dispatched before a Document is saved under a new name. This event bubbles. This event is cancelable. |

DocumentEvents

Object available in ID 7

| DOCUMENTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|---|
| afterActivate | <code>const afterActivate(afterActivate)</code> | Dispatched after the Document becomes active. This event bubbles. This event is not cancelable. |
| afterExport | <code>const afterExport(afterExport)</code> | Dispatched after a Document is exported. This event bubbles. This event is not cancelable. |
| afterImport | <code>const afterImport(afterImport)</code> | Dispatched after importing a file into a Document. This event bubbles. This event is not cancelable. |
| afterLinksChanged | <code>const afterLinksChanged(afterLinksChanged)</code> | Dispatched after one or more links in the Document have been added, deleted, or modified. This event bubbles. This event is not cancelable. |
| afterNew | <code>const afterNew(afterNew)</code> | Dispatched after a Document is created. This event bubbles. This event is not cancelable. |
| afterOpen | <code>const afterOpen(afterOpen)</code> | Dispatched after a Document is opened. This event bubbles. This event is not cancelable. |
| afterPrint | <code>const afterPrint(afterPrint)</code> | Dispatched after a Document is printed. This event bubbles. This event is not cancelable. |
| afterRevert | <code>const afterRevert(afterRevert)</code> | Dispatched after a Document is reverted. This event bubbles. This event is not cancelable. |
| afterSave | <code>const afterSave(afterSave)</code> | Dispatched after a Document is saved. This event bubbles. This event is not cancelable. |
| afterSaveACopy | <code>const afterSaveACopy(afterSaveACopy)</code> | Dispatched after a copy of a Document is saved. This event bubbles. This event is not cancelable. |
| afterSaveAs | <code>const afterSaveAs(afterSaveAs)</code> | Dispatched after a Document is saved under a new name. This event bubbles. This event is not cancelable. |

| DOCUMENTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|--|---|
| <code>beforeClose</code> | <code>const beforeClose(beforeClose)</code> | <i>Dispatched before a Document is closed. This event bubbles. This event is not cancelable.</i> |
| <code>beforeDeactivate</code> | <code>const beforeDeactivate(beforeDeactivate)</code> | <i>Dispatched before the Document becomes inactive. This event bubbles. This event is not cancelable.</i> |
| <code>beforeExport</code> | <code>const beforeExport(beforeExport)</code> | <i>Dispatched before a Document is exported. This event bubbles. This event is cancelable.</i> |
| <code>beforeImport</code> | <code>const beforeImport(beforeImport)</code> | <i>Dispatched before importing a file into a Document. This event bubbles. This event is cancelable.</i> |
| <code>beforePrint</code> | <code>const beforePrint(beforePrint)</code> | <i>Dispatched before a Document is printed. This event bubbles. This event is cancelable.</i> |
| <code>beforeRevert</code> | <code>const beforeRevert(beforeRevert)</code> | <i>Dispatched before a Document is reverted. This event bubbles. This event is cancelable.</i> |
| <code>beforeSave</code> | <code>const beforeSave(beforeSave)</code> | <i>Dispatched before a Document is saved. This event bubbles. This event is cancelable.</i> |
| <code>beforeSaveACopy</code> | <code>const beforeSaveACopy(beforeSaveACopy)</code> | <i>Dispatched before a copy of a Document is saved. This event bubbles. This event is cancelable.</i> |
| <code>beforeSaveAs</code> | <code>const beforeSaveAs(beforeSaveAs)</code> | <i>Dispatched before a Document is saved under a new name. This event bubbles. This event is cancelable.</i> |
| <code>failedExport</code> | <code>const failedExport(failedExport)</code> | <i>Dispatched after a Document export is canceled or fails. This event bubbles. This event is not cancelable.</i> |

EPSEvents

Object available in ID 7

| EPSEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|--|--|
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a EPS is placed. This event bubbles. This event is not cancelable.</i> |
| <code>beforePlace</code> | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a EPS is placed. This event bubbles. This event is cancelable.</i> |

EPSText

Object available in ID 7

| EPSTEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|---|--|
| <code>applyObjectStyle</code> | <code>void applyObjectStyle(ObjectStyle Using, boolean ClearingOverrides, boolean ClearingOverridesThroughRootObjectStyle)</code> | <i>Applies the specified object style. / Using: The object style to apply. / ClearingOverrides: If true, clears the EPSText's existing attributes before applying the style. / ClearingOverridesThroughRootObjectStyle: If true, clears attributes and formatting applied to the EPSText that are not defined in the object style.</i> |

| EPSTEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|--|
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| autoTag | <code>void autoTag()</code> | Tag the object or the parent story using default tags defined in XML preference. |
| bringForward | <code>void bringForward()</code> | Brings the EPSText forward one level in its layer. |
| bringToFront | <code>void bringToFront(variant Reference)</code> | Brings the EPSText to the front of its layer or in front of a particular item. / Reference: The reference object to bring the object in front of (must have same parent). Type: PageItem. |
| changeObject | <code>{Objects} changeObject(variant ReverseOrder)</code> | Finds objects that match the find what value and replace the objects with the change to value. / Return value: The replaced object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| clearObjectStyleOverrides | <code>void clearObjectStyleOverrides()</code> | Clear overrides for object style. |
| clearTransformations | <code>void clearTransformations()</code> | Clears transformations from the EPSText. Transformations include rotation, scaling, flipping, fitting, and shearing. |
| convertShape | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | Converts the EPSText to a different shape. / Given: The EPSText's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0). Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points). |
| delete | <code>void delete()</code> | Deletes the EPSText. |
| detach | <code>void detach()</code> | Detaches an overridden master page item from the master page. |
| duplicate | <code>{PageItem} duplicate(variant To, variant By)</code> | Duplicates the EPSText at the specified location or offset. / Return value: The duplicated EPSText. / To: The location of the new EPSText, specified in coordinates in the format [x, y]. Type: Array of 2 Units (Doubles or Strings), Spread, Page or Layer. / By: Amount by which to offset the new EPSText from the original EPSText's position. Type: Array of 2 Units (Doubles or Strings). |
| export | <code>void export(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | Exports the object(s) to a file. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| extractLabel | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| findObject | <code>{Objects} findObject(variant ReverseOrder)</code> | Finds objects that match the find what value. / Return value: The found object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| fit | <code>void fit(#FitOptions Given)</code> | Applies the specified fit option to content in a frame. / Given: The fit option to use. |

| EPSTEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------|---|--|
| flipItem | <code>void flipItem(#Flip Given, variant Around)</code> | Flips the EPSText. / Given: The axis around which to flip the EPSText. / Around: The point around which to flip the EPSText. Type: Array of 2 Units (Doubles or Strings) or idAnchorPoint enumerator. |
| insertLabel | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| markup | <code>void markup(XMLElement Using)</code> | Associates the page item with the specified XML element while preserving existing content. / Using: The XML element. |
| move | <code>void move(variant To, variant By)</code> | Moves the EPSText to a new location. Note: Either the 'to' or 'by' parameter is required; if both parameters are defined, only the to value is used. / To: The new location of the EPSText, in the format (x, y). Type: Array of 2 Units (Doubles or Strings), Spread, Page or Layer. / By: The amount (in measurement units) to move the EPSText relative to its current position, in the format (x, y). Type: Array of 2 Units (Doubles or Strings). |
| override | <code>variant override(Page DestinationPage)</code> | Overrides a master page item and places the item on the document page as a new object. / Return value: The new object. Type: PageItem, Guide, Graphic, Movie or Sound. / DestinationPage: The document page that contains the master page item to override. |
| place | <code>variant place(string FileName, boolean ShowingOptions)</code> | Places the file. / Return value: The placed file. Type: Array of Pageitems, Graphics, Movies, Sounds, Stories or Texts. / FileName: The file to place. / ShowingOptions: Whether to display the import options dialog. |
| placeXML | <code>void placeXML(XMLElement Using)</code> | Places XML content into the specified object. Note: Replaces any existing content. / Using: The XML element whose content you want to place. |
| redefineScaling | <code>void redefineScaling(variant To)</code> | Apply an item's scaling to its content if possible. / To: The scale factors to be left on the item. The default is {1.0, 1.0}. Type: Array of 2 Doubles. |
| reframe | <code>void reframe(variant In, variant OpposingCorners)</code> | Move the bounding box of the page item. / In: The bounding box to resize. Type: idCoordinateSpaces enumerator or Ordered array containing CoordinateSpace:idCoordinateSpaces enumerator, BoundsKind:idBoundingBoxLimits enumerator. / OpposingCorners: Opposing corners of new bounding box in the given coordinate space. Type: Array of 2 Arrays of 2 Doubles. |
| removeOverride | <code>void removeOverride()</code> | Removes the override from a previously overridden master page item. |

| EPSTEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------|--|---|
| resize | <code>void resize(variant In, variant From, #ResizeMethods By, variant Values, boolean ResizeIndividually, boolean ConsideringRulerUnits)</code> | <p>Resize the page item. / In: The bounding box to resize. Type: idCoordinateSpaces enumerator, idBoundingBoxLimits enumerator or Ordered array containing CoordinateSpace:idCoordinateSpaces enumerator, BoundsKind:idBoundingBoxLimits enumerator. / From: The transform origin. Legal specifications: relative to bounding box: anchor {anchor {x,y}, bounds kind [, coordinate space]}; relative to coordinate space: {x,y} {{x,y} [, coordinate space]}; relative to layout window ruler: {{x,y}, page index bounds kind}. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / By: How the current dimensions are affected by the given values. / Values: The width and height values. Legal dimensions specifications: {x, y [, coordinate space]}, {x, resize constraint [, coordinate space]}, or {resize constraint, y [, coordinate space]}; where x and y are real numbers and coordinate space is used to determine _only_ the unit of length for x and y; coordinate space is ignored for the 'current dimensions times' resize method. Type: Array of Doubles, idResizeConstraints enumerators or idCoordinateSpaces enumerators. / ResizeIndividually: If false and multiple page items are targeted, the new dimensions are attained only by moving the individual items rather than resizing them. / ConsideringRulerUnits: If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page.</p> |
| resolve | <code>variant resolve(variant Location, #CoordinateSpaces In, boolean ConsideringRulerUnits)</code> | <p>Get the coordinates of the given location in the specified coordinate system. / Return value: The resolved location for each of the targeted page items. Type: Array of 2 Doubles or Array of Arrays of 2 Doubles. / Location: The location requested. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / In: The coordinate space to use. / ConsideringRulerUnits: If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page.</p> |
| select | <code>void select(#SelectionOptions ExistingSelection)</code> | <p>Selects the object. / ExistingSelection: The selection status of the EPSText in relation to previously selected objects.</p> |
| sendBackward | <code>void sendBackward()</code> | <p>Sends the EPSText back one level in its layer.</p> |
| sendToBack | <code>void sendToBack(variant Reference)</code> | <p>Sends the EPSText to the back of its layer or behind a particular item (must have same parent). / Reference: The reference object to send the object behind. Type: PageItem.</p> |
| store | <code>{Asset} store(Library Using)</code> | <p>Stores the object in the specified library. / Return value: The new library asset(s). / Using: The library in which to store the object.</p> |

| EPSTEXT CSS MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| transform | <code>void transform(#CoordinateSpaces In, variant From, variant WithMatrix, variant ReplacingCurrent, boolean ConsideringRulerUnits)</code> | Transform the page item. / In: The coordinate space to use. / From: The temporary origin during the transformation. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / WithMatrix: Transform matrix. Type: Array of 6 Doubles or TransformationMatrix. / ReplacingCurrent: Transform components to consider; providing this optional parameter causes the target's existing transform components to be replaced with new values. Without this parameter, the given matrix is concatenated onto the target's existing transform combining the effect of the two. Type: idMatrixContent enumerator, Array of idMatrixContent enumerators or Long Integer. / ConsideringRulerUnits: If true then a ruler based origin is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| transformAgain | <code>variant transformAgain()</code> | Transforms the EPSText using the last transformation performed on any object. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformAgainIndividually | <code>variant transformAgainIndividually()</code> | Transforms the EPSText using the last transformation performed on any EPSText. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformSequenceAgain | <code>variant transformSequenceAgain()</code> | Transforms the EPSText using the last sequence of transform operations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformSequenceAgainIndividually | <code>variant transformSequenceAgainIndividually()</code> | Transforms the EPSText using the last sequence of transformations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformValuesOf | <code>{Objects} transformValuesOf(#CoordinateSpaces In)</code> | Get the transformation values of the page item. / Return value: A list of the transformation matrices. / In: The coordinate space to use. |
| absoluteFlip | <code>#Flip absoluteFlip</code> | Indicates whether the EPSText has been flipped independently of its parent object and, if yes, the direction in which the EPSText was flipped. |
| absoluteHorizontalScale | <code>double absoluteHorizontalScale</code> | The horizontal scale of the EPSText relative to its containing object. |
| absoluteRotationAngle | <code>double absoluteRotationAngle</code> | The rotation angle of the EPSText relative to its containing object. (Range: -360 to 360). |
| absoluteShearAngle | <code>double absoluteShearAngle</code> | The skewing angle of the EPSText relative to its containing object. (Range: -360 to 360). |
| absoluteVerticalScale | <code>double absoluteVerticalScale</code> | The vertical scale of the EPSText relative to its containing object. |
| allGraphics | <code>{Objects} *allGraphics</code> | Lists all graphics contained by the EPSText. |
| allowOverrides | <code>boolean allowOverrides</code> | If true, the master page item can be overridden. |
| allPageItems | <code>{Objects} *allPageItems</code> | Lists all page items contained by the EPSText. |
| anchoredObjectSettings | <code>{AnchoredObjectSetting} *anchoredObjectSettings</code> | Anchored object settings. |
| animationSettings | <code>{AnimationSetting} *animationSettings</code> | The page item animation settings. |
| appliedObjectStyle | <code>{ObjectStyle} appliedObjectStyle</code> | The object style applied to the EPSText. |
| associatedXMLElement | <code>{XMLElement} *associatedXMLElement</code> | The XML element associated with the EPSText. |
| bottomLeftCornerOption | <code>#CornerOptions bottomLeftCornerOption</code> | The shape to apply to the bottom left corner of rectangular shapes. |

| EPSTEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|--|
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>contentTransparencySettings</code> | <code>{ContentTransparencySetting} *contentTransparencySettings</code> | <i>Transparency settings for the content of the EPSText.</i> |
| <code>endCap</code> | <code>#EndCap endCap</code> | <i>The end shape of an open path.</i> |
| <code>endJoin</code> | <code>#EndJoin endJoin</code> | <i>The corner join applied to the EPSText.</i> |
| <code>fillColor</code> | <code>variant fillColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of the EPSText. . Type: Swatch or String.</i> |
| <code>fillTint</code> | <code>double fillTint</code> | <i>The percent of tint to use in the EPSText's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>fillTransparencySettings</code> | <code>{FillTransparencySetting} *fillTransparencySettings</code> | <i>Transparency settings for the fill applied to the EPSText.</i> |
| <code>flip</code> | <code>#Flip flip</code> | <i>The direction in which to flip the printed image.</i> |
| <code>gapColor</code> | <code>{Swatch} gapColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type.</i> |
| <code>gapTint</code> | <code>double gapTint</code> | <i>The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>geometricBounds</code> | <code>variant geometricBounds</code> | <i>The bounds of the EPSText excluding the stroke width, in the format [y1, x1, y2, x2], which give the coordinates of the top-left and bottom-right corners of the bounding box. Type: Array of 4 Units (Doubles or Strings).</i> |
| <code>gradientFillAngle</code> | <code>double gradientFillAngle</code> | <i>The angle of a linear gradient applied to the fill of the EPSText. (Range: -180 to 180).</i> |
| <code>gradientFillLength</code> | <code>variant gradientFillLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the EPSText. Type: Unit (Double or String).</i> |
| <code>gradientFillStart</code> | <code>variant gradientFillStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the fill of the EPSText, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>gradientStrokeAngle</code> | <code>double gradientStrokeAngle</code> | <i>The angle of a linear gradient applied to the stroke of the EPSText. (Range: -180 to 180).</i> |
| <code>gradientStrokeLength</code> | <code>variant gradientStrokeLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the EPSText. Type: Unit (Double or String).</i> |
| <code>gradientStrokeStart</code> | <code>variant gradientStrokeStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the stroke of the EPSText, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>horizontalScale</code> | <code>double horizontalScale</code> | <i>The horizontal scaling applied to the EPSText.</i> |
| <code>id</code> | <code>integer *id</code> | <i>The unique ID of the EPSText.</i> |
| <code>index</code> | <code>integer *index</code> | <i>The index of the EPSText within its containing object.</i> |
| <code>itemLayer</code> | <code>{Layer} itemLayer</code> | <i>The layer that the EPSText is on.</i> |
| <code>label</code> | <code>string label</code> | <i>A property that can be set to any string.</i> |
| <code>leftLineEnd</code> | <code>#ArrowHead leftLineEnd</code> | <i>The arrowhead applied to the start of the path.</i> |
| <code>localDisplaySetting</code> | <code>#DisplaySettingOptions localDisplaySetting</code> | <i>Display performance options for the EPSText.</i> |

| EPSTEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| <code>locked</code> | <code>boolean locked</code> | If true, the EPSText is locked. |
| <code>miterLimit</code> | <code>double miterLimit</code> | The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500). |
| <code>name</code> | <code>string name</code> | The name of the EPSText. |
| <code>nonprinting</code> | <code>boolean nonprinting</code> | If true, the EPSText does not print. |
| <code>overprintFill</code> | <code>boolean overprintFill</code> | If true, the EPSText's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors. |
| <code>overprintGap</code> | <code>boolean overprintGap</code> | If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors. |
| <code>overprintStroke</code> | <code>boolean overprintStroke</code> | If true, the EPSText's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors. |
| <code>overridden</code> | <code>boolean *overridden</code> | If true, the object originated on a master spread and was overridden. If false, the object either originated on a master spread and was not overridden, or the object did not originate on a master page. |
| <code>overriddenMasterPageItem</code> | <code>variant *overriddenMasterPageItem</code> | An object that originated on a master page and has been overridden. Type: PageItem, Guide, Graphic, Movie or Sound. |
| <code>parent</code> | <code>variant *parent</code> | The parent of the EPSText (a Snippet, PlaceGun, Spread, MasterSpread, SplineItem, Polygon, GraphicLine, Rectangle, Oval, Group, State or Character). Type: Snippet, PlaceGun, Spread, MasterSpread, SplineItem, Polygon, GraphicLine, Rectangle, Oval, Group, State or Character. |
| <code>parentPage</code> | <code>{Page} *parentPage</code> | The page on which this page item appears. |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| <code>rightLineEnd</code> | <code>#ArrowHead rightLineEnd</code> | The arrowhead applied to the end of the path. |
| <code>rotationAngle</code> | <code>double rotationAngle</code> | The rotation angle of the EPSText. (Range: -360 to 360). |
| <code>shearAngle</code> | <code>double shearAngle</code> | The skewing angle applied to the EPSText. (Range: -360 to 360). |
| <code>strokeAlignment</code> | <code>#StrokeAlignment strokeAlignment</code> | The stroke alignment applied to the EPSText. |
| <code>strokeColor</code> | <code>variant strokeColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the EPSText. Type: Swatch or String. |
| <code>strokeCornerAdjustment</code> | <code>#StrokeCornerAdjustment strokeCornerAdjustment</code> | The corner adjustment applied to the EPSText. |
| <code>strokeDashAndGap</code> | <code>variant strokeDashAndGap</code> | The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings). |
| <code>strokeTint</code> | <code>double strokeTint</code> | The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| <code>strokeTransparencySettings</code> | <code>{StrokeTransparencySetting} *strokeTransparencySettings</code> | Transparency settings for the stroke. |
| <code>strokeType</code> | <code>variant strokeType</code> | The name of the stroke style to apply. Type: StrokeStyle or String. |
| <code>strokeWeight</code> | <code>variant strokeWeight</code> | The weight (in points) to apply to the EPSText's stroke. Type: Unit (Double or String). |
| <code>textPaths</code> | <code>{TextPaths} *textPaths</code> | A collection of text paths. |
| <code>textWrapPreferences</code> | <code>{TextWrapPreference} *textWrapPreferences</code> | The text wrap preference properties that define the default formatting for wrapping text around objects. |

| EPSTEXT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|--|
| <code>timingSettings</code> | <code>{TimingSetting} *timingSettings</code> | <i>The object timing settings.</i> |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>transparencySettings</code> | <code>{TransparencySetting} *transparencySettings</code> | <i>Transparency settings.</i> |
| <code>verticalScale</code> | <code>double verticalScale</code> | <i>The vertical scaling applied to the EPSText.</i> |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the EPSText is visible.</i> |
| <code>visibleBounds</code> | <code>variant visibleBounds</code> | <i>The bounds of the EPSText including the stroke width, in the format [y1, x1, y2, x2], which give the coordinates of the top-left and bottom-right corners of the bounding box. Type: Array of 4 Units (Doubles or Strings).</i> |

EPSTextEvents

Object available in ID 7

| EPSTEXTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|--|
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a EPSText is placed. This event bubbles. This event is not cancelable.</i> |
| <code>beforePlace</code> | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a EPSText is placed. This event bubbles. This event is cancelable.</i> |

EPSTexts

Object available in ID 7

| EPSTEXTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|--|---|
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | <i>Returns any object in the collection. / Return value: The object.</i> |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | <i>Returns the first object in the collection. / Return value: The first object.</i> |
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | <i>Returns the object with the specified ID. / Return value: The object. / ID: The ID.</i> |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | <i>Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String.</i> |
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | <i>Returns the last object in the collection. / Return value: The last object.</i> |

| EPSTEXTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|--|
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| <code>previousItem</code> | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| <code>count</code> | <code>integer *count</code> | Displays the number of elements in the EPSText. |
| <code>item</code> | <code>{EPSText} *item(variant Index)</code> | Returns the object with the specified index or name. Type: Long Integer or String. |

EventEvents

Object available in ID 7

| EVENTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| <code>afterActivate</code> | <code>const afterActivate(afterActivate)</code> | Dispatched after the Event becomes active. This event bubbles. This event is not cancelable. |
| <code>afterClose</code> | <code>const afterClose(afterClose)</code> | Dispatched when a Event is closing. Since the close has been committed, it can no longer be canceled. This event bubbles. This event is not cancelable. |
| <code>afterContextChanged</code> | <code>const afterContextChanged(afterContextChanged)</code> | Dispatched after the active context changes. This event bubbles. This event is not cancelable. |
| <code>afterDelete</code> | <code>const afterDelete(afterDelete)</code> | Dispatched after a Event is deleted. This event bubbles. This event is not cancelable. |
| <code>afterEmbed</code> | <code>const afterEmbed(afterEmbed)</code> | Dispatched after a Event is embedded. This event bubbles. This event is not cancelable. |
| <code>afterInvoke</code> | <code>const afterInvoke(afterInvoke)</code> | Dispatched after the Event is invoked. This event does not bubble. This event is not cancelable. |
| <code>afterLinksChanged</code> | <code>const afterLinksChanged(afterLinksChanged)</code> | Dispatched after one or more links in the Event have been added, deleted, or modified. This event bubbles. This event is not cancelable. |
| <code>afterMove</code> | <code>const afterMove(afterMove)</code> | Dispatched after a Event is relocated from one object to another. This event bubbles. This event is not cancelable. |
| <code>afterNew</code> | <code>const afterNew(afterNew)</code> | Dispatched after a Event is created. This event bubbles. This event is not cancelable. |
| <code>afterOpen</code> | <code>const afterOpen(afterOpen)</code> | Dispatched after a Event is opened. This event bubbles. This event is not cancelable. |
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | Dispatched after a Event is placed. This event bubbles. This event is not cancelable. |
| <code>afterQuit</code> | <code>const afterQuit(afterQuit)</code> | Dispatched when the Event is quitting. Since the quit has been committed, it can not be canceled. This event bubbles. This event is not cancelable. |
| <code>afterSelectionAttributeChanged</code> | <code>const afterSelectionAttributeChanged(afterSelectionAttributeChanged)</code> | Dispatched after an attribute on the active selection changes. This event bubbles. This event is not cancelable. |

| EVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|---|
| afterSelectionChanged | <code>const afterSelectionChanged(afterSelectionChanged)</code> | <i>Dispatched after the active selection changes. This event bubbles. This event is not cancelable.</i> |
| afterUnembed | <code>const afterUnembed(afterUnembed)</code> | <i>Dispatched after a Event is unembedded. This event bubbles. This event is not cancelable.</i> |
| afterUpdate | <code>const afterUpdate(afterUpdate)</code> | <i>Dispatched after a Event is updated. This event bubbles. This event is not cancelable.</i> |
| beforeClose | <code>const beforeClose(beforeClose)</code> | <i>Dispatched before a Event is closed. This event bubbles. This event is cancelable.</i> |
| beforeDeactivate | <code>const beforeDeactivate(beforeDeactivate)</code> | <i>Dispatched before the Event becomes inactive. This event bubbles. This event is not cancelable.</i> |
| beforeDelete | <code>const beforeDelete(beforeDelete)</code> | <i>Dispatched before a Event is deleted. This event bubbles. This event is cancelable.</i> |
| beforeDisplay | <code>const beforeDisplay(beforeDisplay)</code> | <i>Dispatched before the Event is displayed. This event does not bubble. This event is not cancelable.</i> |
| beforeEmbed | <code>const beforeEmbed(beforeEmbed)</code> | <i>Dispatched before a Event is embedded. This event bubbles. This event is cancelable.</i> |
| beforeInvoke | <code>const beforeInvoke(beforeInvoke)</code> | <i>Dispatched before the Event is invoked. This event does not bubble. This event is cancelable.</i> |
| beforeMove | <code>const beforeMove(beforeMove)</code> | <i>Dispatched before a Event is relocated from one object to another. This event bubbles. This event is cancelable.</i> |
| beforePlace | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a Event is placed. This event bubbles. This event is cancelable.</i> |
| beforeQuit | <code>const beforeQuit(beforeQuit)</code> | <i>Dispatched before the Event is quit. Allows the quit to be canceled. This event bubbles. This event is cancelable.</i> |
| beforeUnembed | <code>const beforeUnembed(beforeUnembed)</code> | <i>Dispatched before a Event is unembedded. This event bubbles. This event is cancelable.</i> |
| beforeUpdate | <code>const beforeUpdate(beforeUpdate)</code> | <i>Dispatched before a Event is updated. This event bubbles. This event is cancelable.</i> |
| onInvoke | <code>const onInvoke(onInvoke)</code> | <i>Dispatched when the Event is invoked. This event does not bubble. This event is not cancelable.</i> |

FindChangeTransliterateOption

Object available in ID 7

| FINDCHANGETRANSLITERATEOPTION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| caseSensitive | <code>boolean caseSensitive</code> | <i>If true, finds strings whose use of case matches the find text string. If false, finds strings that match the find text string regardless of case.</i> |
| includeFootnotes | <code>boolean includeFootnotes</code> | <i>If true, includes footnotes in the find/change query.</i> |
| includeHiddenLayers | <code>boolean includeHiddenLayers</code> | <i>If true, includes hidden layers in the find/change query.</i> |
| includeLockedLayersForFind | <code>boolean includeLockedLayersForFind</code> | <i>If true, includes locked layers in the find query.</i> |
| includeLockedStoriesForFind | <code>boolean includeLockedStoriesForFind</code> | <i>If true, includes locked stories in the find query.</i> |

| FINDCHANGETRANSLITERATEOPTION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| <code>includeMasterPages</code> | <code>boolean includeMasterPages</code> | If true, includes master pages in the find/change query. |
| <code>kanaSensitive</code> | <code>boolean kanaSensitive</code> | If true, finds only text that matches the specified kana type. |
| <code>parent</code> | <code>{Application} *parent</code> | The parent of the FindChangeTransliterateOption (a Application). |
| <code>wholeWord</code> | <code>boolean wholeWord</code> | If true, finds only the complete find text string. If false, also finds strings that contain the find text string. |
| <code>widthSensitive</code> | <code>boolean widthSensitive</code> | If true, finds only text that matches the specified character width. |

FindTransliteratePreference

Object available in ID 7

| FINDTRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|---|
| <code>allowArbitraryHyphenation</code> | <code>variant allowArbitraryHyphenation</code> | If true, words unassociated with a hyphenation dictionary can break to the next line on any character. Type: Boolean or idNothingEnum enumerator. |
| <code>appliedCharacterStyle</code> | <code>variant appliedCharacterStyle</code> | The character style to search for or change to. Type: String, CharacterStyle or idNothingEnum enumerator. |
| <code>appliedFont</code> | <code>variant appliedFont</code> | The font applied to the FindTransliteratePreference, specified as either a font object or the name of font family. Type: Font, String or idNothingEnum enumerator. |
| <code>appliedLanguage</code> | <code>variant appliedLanguage</code> | The language to search for or change to. Type: String, Language, LanguageWithVendors or idNothingEnum enumerator. |
| <code>appliedNumberingList</code> | <code>variant appliedNumberingList</code> | The list to be part of. Type: NumberingList, String or idNothingEnum enumerator. |
| <code>appliedParagraphStyle</code> | <code>variant appliedParagraphStyle</code> | The paragraph style to search for or change to. Type: String, ParagraphStyle or idNothingEnum enumerator. |
| <code>autoLeading</code> | <code>variant autoLeading</code> | The percent of the type size to use for auto leading. (Range: 0 to 500). Type: Double or idNothingEnum enumerator. |
| <code>autoTcy</code> | <code>variant autoTcy</code> | The number of half-width characters at or below which the characters automatically run horizontally in vertical text. Type: Integer or idNothingEnum enumerator. |
| <code>autoTcyIncludeRoman</code> | <code>variant autoTcyIncludeRoman</code> | If true, auto tcy includes Roman characters. Type: Boolean or idNothingEnum enumerator. |
| <code>balanceRaggedLines</code> | <code>variant balanceRaggedLines</code> | If true or set to an enumeration value, balances ragged lines. Note: Not valid with a single-line text composer. Type: Boolean, idBalanceLinesStyle enumerator or idNothingEnum enumerator. |
| <code>baselineShift</code> | <code>variant baselineShift</code> | The baseline shift applied to the text. Type: Unit (Double or String) or idNothingEnum enumerator. |
| <code>bulletChar</code> | <code>variant *bulletChar</code> | Bullet character. Type: Bullet or idNothingEnum enumerator. |
| <code>bulletsAlignment</code> | <code>variant bulletsAlignment</code> | The alignment of the bullet character. Type: idListAlignment enumerator or idNothingEnum enumerator. |
| <code>bulletsAndNumberingListType</code> | <code>variant bulletsAndNumberingListType</code> | List type for bullets and numbering. Type: idListType enumerator or idNothingEnum enumerator. |

| FINDTRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| bulletsCharacterStyle | <code>variant bulletsCharacterStyle</code> | <i>The character style to be used for the text after string. Type: CharacterStyle, String or idNothingEnum enumerator.</i> |
| bulletsTextAfter | <code>variant bulletsTextAfter</code> | <i>The text after string expression for bullets. Type: String or idNothingEnum enumerator.</i> |
| bunriKinshi | <code>variant bunriKinshi</code> | <i>If true, adds the double period (..), ellipse (...), and double hyphen (--) to the selected kinsoku set. Note: Valid only when a kinsoku set is in effect. Type: Boolean or idNothingEnum enumerator.</i> |
| capitalization | <code>variant capitalization</code> | <i>The capitalization scheme. Type: idCapitalization enumerator or idNothingEnum enumerator.</i> |
| characterAlignment | <code>variant characterAlignment</code> | <i>The alignment of small characters to the largest character in the line. Type: idCharacterAlignment enumerator or idNothingEnum enumerator.</i> |
| characterRotation | <code>variant characterRotation</code> | <i>The rotation angle (in degrees) of individual characters. Note: The rotation is counterclockwise. Type: Double or idNothingEnum enumerator.</i> |
| cjkGridTracking | <code>variant cjkGridTracking</code> | <i>If true, uses grid tracking to track non-Roman characters in CJK grids. Type: Boolean or idNothingEnum enumerator.</i> |
| composer | <code>variant composer</code> | <i>The text composer to use to compose the text. Type: String or idNothingEnum enumerator.</i> |
| dropCapCharacters | <code>variant dropCapCharacters</code> | <i>The number of characters to drop cap. Type: Integer (0 - 150) or idNothingEnum enumerator.</i> |
| dropcapDetail | <code>variant dropcapDetail</code> | <i>Details about the drop cap based on the glyph outlines. 1 = left side bearing. 2 = descenders. 0x100,0x200,0x400 are used for Japanese frame grid. Type: Long Integer or idNothingEnum enumerator.</i> |
| dropCapLines | <code>variant dropCapLines</code> | <i>The number of lines to drop cap. Type: Integer (0 - 25) or idNothingEnum enumerator.</i> |
| fillColor | <code>variant fillColor</code> | <i>The swatch (color, gradient, tint, or mixed ink), applied as a fill color, to search for or change to. Type: String, Swatch or idNothingEnum enumerator.</i> |
| fillTint | <code>variant fillTint</code> | <i>The tint (as a percentage) of the fill color of the FindTransliteratePreference. (To specify a tint percentage, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). Type: Double or idNothingEnum enumerator.</i> |
| findCharacterType | <code>variant findCharacterType</code> | <i>The character type to find. Type: idFindChangeTransliterateCharacterTypes enumerator or idNothingEnum enumerator.</i> |
| firstLineIndent | <code>variant firstLineIndent</code> | <i>The amount to indent the first line. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| fontStyle | <code>variant fontStyle</code> | <i>The name of the font style. Type: String or idNothingEnum enumerator.</i> |
| glyphForm | <code>variant glyphForm</code> | <i>The glyph variant to substitute for standard glyphs. Type: idAlternateGlyphForms enumerator or idNothingEnum enumerator.</i> |
| gradientFillAngle | <code>variant gradientFillAngle</code> | <i>The angle of a linear gradient applied to the fill of the text. (Range: -180 to 180). Type: Double or idNothingEnum enumerator.</i> |
| gradientFillLength | <code>variant gradientFillLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the text. Type: Double or idNothingEnum enumerator.</i> |
| gradientFillStart | <code>variant gradientFillStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the fill of the text, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings) or idNothingEnum enumerator.</i> |
| gradientStrokeAngle | <code>variant gradientStrokeAngle</code> | <i>The angle of a linear gradient applied to the stroke of the text. (Range: -180 to 180). Type: Double or idNothingEnum enumerator.</i> |

| FINDTRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--------------------------------|--|
| gradientStrokeLength | variant gradientStrokeLength | The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the text. Type: Double or idNothingEnum enumerator. |
| gradientStrokeStart | variant gradientStrokeStart | The starting point (in page coordinates) of a gradient applied to the stroke of the text, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings) or idNothingEnum enumerator. |
| gridAlignment | variant gridAlignment | The alignment to the frame grid or baseline grid. Type: idGridAlignment enumerator or idNothingEnum enumerator. |
| gridGyoudori | variant gridGyoudori | The manual gyoudori setting. Type: Integer or idNothingEnum enumerator. |
| horizontalScale | variant horizontalScale | The horizontal scaling applied to the FindTransliteratePreference. Type: Double or idNothingEnum enumerator. |
| hyphenateAcrossColumns | variant hyphenateAcrossColumns | If true, allows the last word in a text column to be hyphenated. Type: Boolean or idNothingEnum enumerator. |
| hyphenateLastWord | variant hyphenateLastWord | If true, allows hyphenation in the last word in a paragraph. Note: Valid only when hyphenation is true. Type: Boolean or idNothingEnum enumerator. |
| ignoreEdgeAlignment | variant ignoreEdgeAlignment | If true, ignores optical edge alignment for the paragraph. Type: Boolean or idNothingEnum enumerator. |
| jidori | variant jidori | The number of grid squares in which to arrange the text. . Type: Integer or idNothingEnum enumerator. |
| justification | variant justification | The paragraph alignment. Type: idJustification enumerator or idNothingEnum enumerator. |
| keepAllLinesTogether | variant keepAllLinesTogether | If true, keeps all lines of the paragraph together. If false, allows paragraphs to break across pages or columns. Type: Boolean or idNothingEnum enumerator. |
| keepFirstLines | variant keepFirstLines | The minimum number of lines to keep together in a paragraph before allowing a page break. Type: Integer (1 - 50) or idNothingEnum enumerator. |
| keepLastLines | variant keepLastLines | The minimum number of lines to keep together in a paragraph after a page break. Type: Integer (1 - 50) or idNothingEnum enumerator. |
| keepLinesTogether | variant keepLinesTogether | If true, keeps a specified number of lines together when the paragraph breaks across columns or text frames. Type: Boolean or idNothingEnum enumerator. |
| keepRuleAboveInFrame | variant keepRuleAboveInFrame | If true, forces the rule above the paragraph to remain in the frame bounds. Note: Valid only when rule above is true. Type: Boolean or idNothingEnum enumerator. |
| keepWithNext | variant keepWithNext | The minimum number of lines to keep with the next paragraph. Type: Integer (0 - 5) or idNothingEnum enumerator. |
| kentenAlignment | variant kentenAlignment | The alignment of kenten characters relative to the parent characters. . Type: idKntenAlignment enumerator or idNothingEnum enumerator. |
| kentenCharacterSet | variant kentenCharacterSet | The character set used for the custom kenten character. Note: Valid only when kenten kind is custom. . Type: idKntenCharacterSet enumerator or idNothingEnum enumerator. |
| kentenCustomCharacter | variant kentenCustomCharacter | The character used for kenten. Note: Valid only when kenten kind is custom. Type: String or idNothingEnum enumerator. |
| kentenFillColor | variant kentenFillColor | The swatch (color, gradient, tint, or mixed ink) applied to the fill of kenten characters. Type: Swatch, String or idNothingEnum enumerator. |
| kentenFont | variant kentenFont | The font to use for kenten characters. Type: Font, String or idNothingEnum enumerator. |

| FINDTRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|------------------------------|---|
| kntenFontSize | variant kntenFontSize | The size (in points) of knten characters. Type: Double or idNothingEnum enumerator. |
| kntenFontStyle | variant kntenFontStyle | The font style of knten characters. Type: String or idNothingEnum enumerator. |
| kntenKind | variant kntenKind | The style of knten characters. Type: idKntenCharacter enumerator or idNothingEnum enumerator. |
| kntenOverprintFill | variant kntenOverprintFill | The method of overprinting the knten fill. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator. |
| kntenOverprintStroke | variant kntenOverprintStroke | The method of overprinting the knten stroke. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator. |
| kntenPlacement | variant kntenPlacement | The distance between knten characters and their parent characters. Type: Double or idNothingEnum enumerator. |
| kntenPosition | variant kntenPosition | The knten position relative to the parent character. Type: idRubyKetenPosition enumerator or idNothingEnum enumerator. |
| kntenStrokeColor | variant kntenStrokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of knten characters. Type: Swatch, String or idNothingEnum enumerator. |
| kntenStrokeTint | variant kntenStrokeTint | The stroke tint (as a percentage) of knten characters. (Range: 0 to 100). Type: Double or idNothingEnum enumerator. |
| kntenTint | variant kntenTint | The fill tint (as a percentage) of knten characters. (Range: 0 to 100). Type: Double or idNothingEnum enumerator. |
| kntenWeight | variant kntenWeight | The stroke weight (in points) of knten characters. Type: Double or idNothingEnum enumerator. |
| kntenXScale | variant kntenXScale | The horizontal size of knten characters as a percent of the original size. Type: Double or idNothingEnum enumerator. |
| kntenYScale | variant kntenYScale | The vertical size of knen characters as a percent of the original size. . Type: Double or idNothingEnum enumerator. |
| kerningMethod | variant kerningMethod | The type of pair kerning. Type: String or idNothingEnum enumerator. |
| kerningValue | variant kerningValue | The amount of space to add or remove between characters, specified in thousands of an em. . Type: Double or idNothingEnum enumerator. |
| kinsokuHangType | variant kinsokuHangType | The type of hanging punctuation to allow. Note: Valid only when a kinsoku set is in effect. Type: idKinsokuHangTypes enumerator or idNothingEnum enumerator. |
| kinsokuSet | variant kinsokuSet | The kinsoku set that determines legitimate line breaks. Type: KinsokuTable, idKinsokuSet enumerator, String or idNothingEnum enumerator. |
| kinsokuType | variant kinsokuType | The type of kinsoku processing for preventing kinsoku characters from beginning or ending a line. Note: Valid only when a kinsoku set is defined. Type: idKinsokuType enumerator or idNothingEnum enumerator. |
| lastLineIndent | variant lastLineIndent | The amount to indent the last line in the paragraph. Type: Unit (Double or String) or idNothingEnum enumerator. |
| leading | variant leading | The leading applied to the text. Type: Unit (Double or String), idLeading enumerator or idNothingEnum enumerator. |
| leadingAki | variant leadingAki | The amount of space before each character. Type: Double or idNothingEnum enumerator. |
| leadingModel | variant leadingModel | The point from which leading is measured from line to line. Type: idLeadingModel enumerator or idNothingEnum enumerator. |

| FINDTRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| leftIndent | <code>variant leftIndent</code> | <i>The width of the left indent. Type: Unit (Double or String) or idNothingEnum enumerator.</i> |
| ligatures | <code>variant ligatures</code> | <i>If true, replaces specific character combinations (e.g., fl, fi) with ligature characters. Type: Boolean or idNothingEnum enumerator.</i> |
| mojikumi | <code>variant mojikumi</code> | <i>The mojikumi table. For information, see mojikumi table defaults. Type: MojikumiTable, String, idMojikumiTableDefaults enumerator or idNothingEnum enumerator.</i> |
| noBreak | <code>variant noBreak</code> | <i>If true, keeps the text on the same line. . Type: Boolean or idNothingEnum enumerator.</i> |
| numberingAlignment | <code>variant numberingAlignment</code> | <i>The alignment of the number. Type: idListAlignment enumerator or idNothingEnum enumerator.</i> |
| numberingApplyRestartPolicy | <code>variant numberingApplyRestartPolicy</code> | <i>If true, apply the numbering restart policy. Type: Boolean or idNothingEnum enumerator.</i> |
| numberingCharacterStyle | <code>variant numberingCharacterStyle</code> | <i>The character style to be used for the number string. Type: CharacterStyle, String or idNothingEnum enumerator.</i> |
| numberingContinue | <code>variant numberingContinue</code> | <i>Continue the numbering at this level. Type: Boolean or idNothingEnum enumerator.</i> |
| numberingExpression | <code>variant numberingExpression</code> | <i>The number string expression for numbering. Type: String or idNothingEnum enumerator.</i> |
| numberingFormat | <code>variant numberingFormat</code> | <i>Numbering format options. Type: idNumberingStyle enumerator, String or idNothingEnum enumerator.</i> |
| numberingLevel | <code>variant numberingLevel</code> | <i>The level of the paragraph. Type: Long Integer or idNothingEnum enumerator.</i> |
| numberingStartAt | <code>variant numberingStartAt</code> | <i>Determines starting number in a numbered list. Type: Long Integer or idNothingEnum enumerator.</i> |
| otfContextualAlternate | <code>variant otfContextualAlternate</code> | <i>If true, uses contextual alternate forms in OpenType fonts. Type: Boolean or idNothingEnum enumerator.</i> |
| otfDiscretionaryLigature | <code>variant otfDiscretionaryLigature</code> | <i>If true, uses discretionary ligatures in OpenType fonts. Type: Boolean or idNothingEnum enumerator.</i> |
| otfFigureStyle | <code>variant otfFigureStyle</code> | <i>The figure style in OpenType fonts. Type: idOTFFigureStyle enumerator or idNothingEnum enumerator.</i> |
| otfFraction | <code>variant otfFraction</code> | <i>If true, uses fractions in OpenType fonts. Type: Boolean or idNothingEnum enumerator.</i> |
| otfHistorical | <code>variant otfHistorical</code> | <i>If true, use historical forms in OpenType fonts. Type: Boolean or idNothingEnum enumerator.</i> |
| otfHVKana | <code>variant otfHVKana</code> | <i>If true, switches hiragana fonts, which have different glyphs for horizontal and vertical. . Type: Boolean or idNothingEnum enumerator.</i> |
| otfLocale | <code>variant otfLocale</code> | <i>If true, uses localized forms in OpenType fonts. Type: Boolean or idNothingEnum enumerator.</i> |
| otfMark | <code>variant otfMark</code> | <i>If true, uses mark positioning in OpenType fonts. Type: Boolean or idNothingEnum enumerator.</i> |
| otfOrdinal | <code>variant otfOrdinal</code> | <i>If true, uses ordinals in OpenType fonts. Type: Boolean or idNothingEnum enumerator.</i> |
| otfProportionalMetrics | <code>variant otfProportionalMetrics</code> | <i>If true, kerns according to proportional CJK metrics in OpenType fonts. Type: Boolean or idNothingEnum enumerator.</i> |

| FINDTRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|---|
| otfRomanItalics | <code>variant otfRomanItalics</code> | If true, applies italics to half-width alphanumeric. . Type: Boolean or idNothingEnum enumerator. |
| otfSlashedZero | <code>variant otfSlashedZero</code> | If true, use a slashed zeroes in OpenType fonts. Type: Boolean or idNothingEnum enumerator. |
| otfStylisticSets | <code>variant otfStylisticSets</code> | The stylistic sets to use in OpenType fonts. Type: Long Integer or idNothingEnum enumerator. |
| otfSwash | <code>variant otfSwash</code> | If true, uses swash forms in OpenType fonts. Type: Boolean or idNothingEnum enumerator. |
| otfTitling | <code>variant otfTitling</code> | If true, uses titling forms in OpenType fonts. Type: Boolean or idNothingEnum enumerator. |
| overprintFill | <code>variant overprintFill</code> | If true, the fill color of the characters will overprint. Type: Boolean or idNothingEnum enumerator. |
| overprintStroke | <code>variant overprintStroke</code> | If true, the stroke of the characters will overprint. Type: Boolean or idNothingEnum enumerator. |
| paragraphGyoudori | <code>variant paragraphGyoudori</code> | If true, the gyoudori mode applies to the entire paragraph. If false, the gyoudori mode applies to each line in the paragraph. Type: Boolean or idNothingEnum enumerator. |
| parent | <code>{Application} *parent</code> | The parent of the FindTransliteratePreference (a Application). |
| pointSize | <code>variant pointSize</code> | The text size. Type: Unit (Double or String) or idNothingEnum enumerator. |
| position | <code>variant position</code> | The text position relative to the baseline. Type: idPosition enumerator or idNothingEnum enumerator. |
| positionalForm | <code>variant positionalForm</code> | The OpenType positional form. Type: idPositionalForms enumerator or idNothingEnum enumerator. |
| preferences | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| rensuumi | <code>variant rensuumi</code> | If true, disallows line breaks in numbers. If false, lines can break between digits in multi-digit numbers. Type: Boolean or idNothingEnum enumerator. |
| rightIndent | <code>variant rightIndent</code> | The width of the right indent. Type: Unit (Double or String) or idNothingEnum enumerator. |
| rotateSingleByteCharacters | <code>variant rotateSingleByteCharacters</code> | If true, rotates Roman characters in vertical text. Type: Boolean or idNothingEnum enumerator. |
| rubyAlignment | <code>variant rubyAlignment</code> | The ruby alignment. Type: idRubyAlignments enumerator or idNothingEnum enumerator. |
| rubyAutoAlign | <code>variant rubyAutoAlign</code> | If true, auto aligns ruby. Type: Boolean or idNothingEnum enumerator. |
| rubyAutoScaling | <code>variant rubyAutoScaling</code> | If true, automatically scales ruby to the specified percent of parent text size. For information on specifying a percent, see ruby parent scaling percent. . Type: Boolean or idNothingEnum enumerator. |
| rubyAutoTcyAutoScale | <code>variant rubyAutoTcyAutoScale</code> | If true, automatically scales glyphs in auto tcy (tate-chuu-yoko) in ruby to fit one em. Type: Boolean or idNothingEnum enumerator. |
| rubyAutoTcyDigits | <code>variant rubyAutoTcyDigits</code> | The number of digits included in auto tcy (tate-chuu-yoko) in ruby. Type: Integer or idNothingEnum enumerator. |
| rubyAutoTcyIncludeRoman | <code>variant rubyAutoTcyIncludeRoman</code> | If true, includes Roman characters in auto tcy (tate-chuu-yoko) in ruby. Type: Boolean or idNothingEnum enumerator. |
| rubyFill | <code>variant rubyFill</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of ruby characters. Type: Swatch, String or idNothingEnum enumerator. |

| FINDTRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| rubyFont | <code>variant rubyFont</code> | <i>The font applied to ruby characters. Type: Font, String or idNothingEnum enumerator.</i> |
| rubyFontSize | <code>variant rubyFontSize</code> | <i>The size (in points) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyFontStyle | <code>variant rubyFontStyle</code> | <i>The font style of ruby characters. Type: String or idNothingEnum enumerator.</i> |
| rubyOpenTypePro | <code>variant rubyOpenTypePro</code> | <i>If true, uses OpenType Pro fonts for ruby. Type: Boolean or idNothingEnum enumerator.</i> |
| rubyOverhang | <code>variant rubyOverhang</code> | <i>If true, constrains ruby overhang to the specified amount. For information on specifying an amount, see ruby parent overhang amount. . Type: Boolean or idNothingEnum enumerator.</i> |
| rubyOverprintFill | <code>variant rubyOverprintFill</code> | <i>The method of overprinting the ruby fill. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator.</i> |
| rubyOverprintStroke | <code>variant rubyOverprintStroke</code> | <i>The method of overprinting the ruby stroke. Type: idAdornmentOverprint enumerator or idNothingEnum enumerator.</i> |
| rubyParentOverhangAmount | <code>variant rubyParentOverhangAmount</code> | <i>The amount by which ruby characters can overhang the parent text. Type: idRubyOverhang enumerator or idNothingEnum enumerator.</i> |
| rubyParentScalingPercent | <code>variant rubyParentScalingPercent</code> | <i>The amount (as a percentage) to scale the parent text size to determine the ruby text size. Type: Double or idNothingEnum enumerator.</i> |
| rubyParentSpacing | <code>variant rubyParentSpacing</code> | <i>The ruby spacing relative to the parent text. . Type: idRubyParentSpacing enumerator or idNothingEnum enumerator.</i> |
| rubyPosition | <code>variant rubyPosition</code> | <i>The position of ruby characters relative to the parent text. Type: idRubyKetenPosition enumerator or idNothingEnum enumerator.</i> |
| rubyStroke | <code>variant rubyStroke</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of ruby characters. Type: Swatch, String or idNothingEnum enumerator.</i> |
| rubyStrokeTint | <code>variant rubyStrokeTint</code> | <i>The stroke tint (as a percentage) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyTint | <code>variant rubyTint</code> | <i>The tint (as a percentage) of the ruby fill color. (Range: 0 to 100). Type: Double or idNothingEnum enumerator.</i> |
| rubyType | <code>variant rubyType</code> | <i>The ruby type. Type: idRubyTypes enumerator or idNothingEnum enumerator.</i> |
| rubyWeight | <code>variant rubyWeight</code> | <i>The stroke weight (in points) of ruby characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyXOffset | <code>variant rubyXOffset</code> | <i>The amount of horizontal space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyXScale | <code>variant rubyXScale</code> | <i>The horizontal size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| rubyYOffset | <code>variant rubyYOffset</code> | <i>The amount of vertical space between ruby and parent characters. Type: Double or idNothingEnum enumerator.</i> |
| rubyYScale | <code>variant rubyYScale</code> | <i>The vertical size of ruby characters, specified as a percent of the original size. Type: Double or idNothingEnum enumerator.</i> |
| scaleAffectsLineHeight | <code>variant scaleAffectsLineHeight</code> | <i>If true, the line changes size when characters are scaled. . Type: Boolean or idNothingEnum enumerator.</i> |
| shataiAdjustRotation | <code>variant shataiAdjustRotation</code> | <i>If true, applies shatai rotation. Type: Boolean or idNothingEnum enumerator.</i> |

| FINDTRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>shataiAdjustTsume</code> | <code>variant shataiAdjustTsume</code> | If true, adjusts shatai tsume. Type: Boolean or idNothingEnum enumerator. |
| <code>shataiDegreeAngle</code> | <code>variant shataiDegreeAngle</code> | The shatai lens angle (in degrees). Type: Double or idNothingEnum enumerator. |
| <code>shataiMagnification</code> | <code>variant shataiMagnification</code> | The amount (as a percentage) of shatai obliquing to apply. Type: Double or idNothingEnum enumerator. |
| <code>singleWordJustification</code> | <code>variant singleWordJustification</code> | The alignment to use for lines that contain a single word. Type: idSingleWordJustification enumerator or idNothingEnum enumerator. |
| <code>skew</code> | <code>variant skew</code> | The skew angle of the FindTransliteratePreference. Type: Double or idNothingEnum enumerator. |
| <code>spaceAfter</code> | <code>variant spaceAfter</code> | The height of the paragraph space below. Type: Unit (Double or String) or idNothingEnum enumerator. |
| <code>spaceBefore</code> | <code>variant spaceBefore</code> | The height of the paragraph space above. Type: Unit (Double or String) or idNothingEnum enumerator. |
| <code>startParagraph</code> | <code>variant startParagraph</code> | The location at which to start the paragraph. Type: idStartParagraph enumerator or idNothingEnum enumerator. |
| <code>strikeThroughColor</code> | <code>variant strikeThroughColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the strikethrough stroke. Type: Swatch, String or idNothingEnum enumerator. |
| <code>strikeThroughGapColor</code> | <code>variant strikeThroughGapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of the strikethrough stroke. Type: Swatch, String or idNothingEnum enumerator. |
| <code>strikeThroughGapOverprint</code> | <code>variant strikeThroughGapOverprint</code> | If true, the gap color of the strikethrough stroke will overprint. Note: Valid when strike through type is not solid. Type: Boolean or idNothingEnum enumerator. |
| <code>strikeThroughGapTint</code> | <code>variant strikeThroughGapTint</code> | The tint (as a percentage) of the strikethrough stroke gap color. (Range: 0 to 100) Note: Valid when strike through type is not solid. Type: Double or idNothingEnum enumerator. |
| <code>strikeThroughOffset</code> | <code>variant strikeThroughOffset</code> | The amount by which to offset the strikethrough stroke from the text baseline. Type: Unit (Double or String) or idNothingEnum enumerator. |
| <code>strikeThroughOverprint</code> | <code>variant strikeThroughOverprint</code> | If true, the strikethrough stroke will overprint. Type: Boolean or idNothingEnum enumerator. |
| <code>strikeThroughTint</code> | <code>variant strikeThroughTint</code> | The tint (as a percentage) of the strikethrough stroke. (Range: 0 to 100). Type: Double or idNothingEnum enumerator. |
| <code>strikeThroughType</code> | <code>variant strikeThroughType</code> | The stroke type of the strikethrough stroke. Type: StrokeStyle, String or idNothingEnum enumerator. |
| <code>strikeThroughWeight</code> | <code>variant strikeThroughWeight</code> | The stroke weight of the strikethrough stroke. Type: Unit (Double or String) or idNothingEnum enumerator. |
| <code>strikeThru</code> | <code>variant strikeThru</code> | If true, draws a strikethrough line through the text. Type: Boolean or idNothingEnum enumerator. |
| <code>strokeColor</code> | <code>variant strokeColor</code> | The swatch (color, gradient, tint, or mixed ink), applied as a stroke color, to search for or change to. Type: String, Swatch or idNothingEnum enumerator. |
| <code>strokeTint</code> | <code>variant strokeTint</code> | The tint (as a percentage) of the stroke color of the FindTransliteratePreference. (To specify a tint percentage, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). Type: Double or idNothingEnum enumerator. |
| <code>strokeWeight</code> | <code>variant strokeWeight</code> | The stroke weight applied to the characters of the text. Type: Unit (Double or String) or idNothingEnum enumerator. |

| FINDTRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|---|
| tatechuyoko | variant <code>tatechuyoko</code> | If true, makes the character horizontal in vertical text. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| tatechuyokoXOffset | variant <code>tatechuyokoXOffset</code> | The horizontal offset for horizontal characters in vertical text. Type: Double or <code>idNothingEnum</code> enumerator. |
| tatechuyokoYOffset | variant <code>tatechuyokoYOffset</code> | The vertical offset for horizontal characters in vertical text. Type: Double or <code>idNothingEnum</code> enumerator. |
| tracking | variant <code>tracking</code> | The amount by which to loosen or tighten a block of text, specified in thousands of an em. Type: Double or <code>idNothingEnum</code> enumerator. |
| trailingAki | variant <code>trailingAki</code> | The amount of space after each character. Type: Double or <code>idNothingEnum</code> enumerator. |
| treatIdeographicSpaceAsSpace | variant <code>treatIdeographicSpaceAsSpace</code> | If true, ideographic spaces will not wrap to the next line like text characters. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| tsume | variant <code>tsume</code> | The amount of horizontal character compression. Type: Double or <code>idNothingEnum</code> enumerator. |
| underline | variant <code>underline</code> | If true, underlines the text. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| underlineColor | variant <code>underlineColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the underline stroke. . Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| underlineGapColor | variant <code>underlineGapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of the underline stroke. Note: Valid when underline type is not solid. Type: Swatch, String or <code>idNothingEnum</code> enumerator. |
| underlineGapOverprint | variant <code>underlineGapOverprint</code> | If true, the gap color of the underline stroke will overprint. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| underlineGapTint | variant <code>underlineGapTint</code> | The tint (as a percentage) of the gap color of the underline stroke. (Range: 0 to 100) Note: Valid when underline type is not solid. Type: Double or <code>idNothingEnum</code> enumerator. |
| underlineOffset | variant <code>underlineOffset</code> | The amount by which to offset the underline from the text baseline. Type: Unit (Double or String) or <code>idNothingEnum</code> enumerator. |
| underlineOverprint | variant <code>underlineOverprint</code> | If true, the underline stroke color will overprint. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| underlineTint | variant <code>underlineTint</code> | The underline stroke tint (as a percentage). (Range: 0 to 100). Type: Double or <code>idNothingEnum</code> enumerator. |
| underlineType | variant <code>underlineType</code> | The stroke type of the underline stroke. Type: <code>StrokeStyle</code> , String or <code>idNothingEnum</code> enumerator. |
| underlineWeight | variant <code>underlineWeight</code> | The stroke weight of the underline stroke. Type: Unit (Double or String) or <code>idNothingEnum</code> enumerator. |
| verticalScale | variant <code>verticalScale</code> | The vertical scaling applied to the <code>FindTransliteratePreference</code> . Type: Double or <code>idNothingEnum</code> enumerator. |
| warichu | variant <code>warichu</code> | If true, turns on warichu. Type: Boolean or <code>idNothingEnum</code> enumerator. |
| warichuAlignment | variant <code>warichuAlignment</code> | The warichu alignment. Type: <code>idWarichuAlignment</code> enumerator or <code>idNothingEnum</code> enumerator. |
| warichuCharsAfterBreak | variant <code>warichuCharsAfterBreak</code> | The minimum number of characters allowed after a line break. . Type: Integer or <code>idNothingEnum</code> enumerator. |
| warichuCharsBeforeBreak | variant <code>warichuCharsBeforeBreak</code> | The minimum number of characters allowed before a line break. Type: Integer or <code>idNothingEnum</code> enumerator. |

| FINDTRANSLITERATEPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|---|--|
| <code>warichuLines</code> | <code>variant warichuLines</code> | <i>The number of lines of warichu within a single normal line. Type: Integer or idNothingEnum enumerator.</i> |
| <code>warichuLineSpacing</code> | <code>variant warichuLineSpacing</code> | <i>The gap between lines of warichu characters. Type: Double or idNothingEnum enumerator.</i> |
| <code>warichuSize</code> | <code>variant warichuSize</code> | <i>The amount (as a percentage) to scale parent text size to determine warichu size. Type: Double or idNothingEnum enumerator.</i> |

FormFieldEvents

Object available in ID 7

| FORMFIELDEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|--|--|
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a FormField is placed. This event bubbles. This event is not cancelable.</i> |
| <code>beforePlace</code> | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a FormField is placed. This event bubbles. This event is cancelable.</i> |

GotoNextStateBehavior

Object available in ID 7

| GOTONEXTSTATEBEHAVIOR CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| <code>delete</code> | <code>void delete()</code> | <i>Deletes the GotoNextStateBehavior.</i> |
| <code>extractLabel</code> | <code>string extractLabel(string Key)</code> | <i>Gets the label value associated with the specified key. / Return value: The label. / Key: The key.</i> |
| <code>insertLabel</code> | <code>void insertLabel(string Key, string Value)</code> | <i>Sets the label to the value associated with the specified key. / Key: The key. / Value: The value.</i> |
| <code>associatedMultiStateObject</code> | <code>{MultiStateObject} associatedMultiStateObject</code> | <i>The associated multi-state object page item.</i> |
| <code>behaviorEvent</code> | <code>#BehaviorEvents behaviorEvent</code> | <i>The event which triggers the behavior.</i> |
| <code>enableBehavior</code> | <code>boolean enableBehavior</code> | <i>If true, the behavior is enabled.</i> |
| <code>id</code> | <code>integer •id</code> | <i>The unique ID of the GotoNextStateBehavior.</i> |
| <code>index</code> | <code>integer •index</code> | <i>The index of the GotoNextStateBehavior within its containing object.</i> |
| <code>label</code> | <code>string label</code> | <i>A property that can be set to any string.</i> |
| <code>loopsToNextOrPrevious</code> | <code>boolean loopsToNextOrPrevious</code> | <i>If true, will loop to the next or previous state.</i> |
| <code>name</code> | <code>string •name</code> | <i>The name of the GotoNextStateBehavior.</i> |
| <code>parent</code> | <code>{Button} •parent</code> | <i>The parent of the GotoNextStateBehavior (a Button).</i> |

GotoNextStateBehaviors

Object available in ID 7

| GOTONEXTSTATEBEHAVIORS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|--|
| <code>add</code> | <code>{GotoNextStateBehavior} add()</code> | Creates a new GotoNextStateBehavior. / Return value: The new GotoNextStateBehavior. |
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | Returns any object in the collection. / Return value: The object. |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | Returns the first object in the collection. / Return value: The first object. |
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | Returns the object with the specified ID. / Return value: The object. / ID: The ID. |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| <code>previousItem</code> | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| <code>count</code> | <code>integer •count</code> | Displays the number of elements in the GotoNextStateBehavior. |
| <code>item</code> | <code>{GotoNextStateBehavior} •item(variant Index)</code> | Returns the object with the specified index or name. Type: Long Integer or String. |

GotoPreviousStateBehavior

Object available in ID 7

| GOTOPREVIOUSSTATEBEHAVIOR CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>delete</code> | <code>void delete()</code> | Deletes the GotoPreviousStateBehavior. |
| <code>extractLabel</code> | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| <code>insertLabel</code> | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| <code>associatedMultiStateObject</code> | <code>{MultiStateObject} associatedMultiStateObject</code> | The associated multi-state object page item. |
| <code>behaviorEvent</code> | <code>#BehaviorEvents behaviorEvent</code> | The event which triggers the behavior. |
| <code>enableBehavior</code> | <code>boolean enableBehavior</code> | If true, the behavior is enabled. |
| <code>id</code> | <code>integer •id</code> | The unique ID of the GotoPreviousStateBehavior. |
| <code>index</code> | <code>integer •index</code> | The index of the GotoPreviousStateBehavior within its containing object. |
| <code>label</code> | <code>string label</code> | A property that can be set to any string. |

| GOTOPREVIOUSSTATEBEHAVIOR CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|--|---|
| loopsToNextOrPrevious | <code>boolean loopsToNextOrPrevious</code> | If true, will loop to the next or previous state. |
| name | <code>string *name</code> | The name of the GotoPreviousStateBehavior. |
| parent | <code>{Button} *parent</code> | The parent of the GotoPreviousStateBehavior (a Button). |

GotoPreviousStateBehaviors

Object available in ID 7

| GOTOPREVIOUSSTATEBEHAVIORS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|---|--|
| add | <code>{GotoPreviousStateBehavior} add()</code> | Creates a new GotoPreviousStateBehavior. / Return value: The new GotoPreviousStateBehavior. |
| anyItem | <code>IDispatch anyItem()</code> | Returns any object in the collection. / Return value: The object. |
| firstItem | <code>IDispatch firstItem()</code> | Returns the first object in the collection. / Return value: The first object. |
| itemByID | <code>IDispatch itemByID(integer Id)</code> | Returns the object with the specified ID. / Return value: The object. / ID: The ID. |
| itemByRange | <code>{Objects} itemByRange(variant From, variant To)</code> | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| lastItem | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |
| middleItem | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| nextItem | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| previousItem | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| count | <code>integer *count</code> | Displays the number of elements in the GotoPreviousStateBehavior. |
| item | <code>{GotoPreviousStateBehavior} *item(variant Index)</code> | Returns the object with the specified index or name. Type: Long Integer or String. |

GotoStateBehavior

Object available in ID 7

| GOTOSTATEBEHAVIOR CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|---|--|
| delete | <code>void delete()</code> | Deletes the GotoStateBehavior. |
| extractLabel | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| insertLabel | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |

| GOTOSTATEBEHAVIOR CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|--|
| associatedMultiStateObject | <code>{MultiStateObject} associatedMultiStateObject</code> | <i>The associated multi-state object page item.</i> |
| behaviorEvent | <code>#BehaviorEvents behaviorEvent</code> | <i>The event which triggers the behavior.</i> |
| enableBehavior | <code>boolean enableBehavior</code> | <i>If true, the behavior is enabled.</i> |
| goBackOnRollOff | <code>boolean goBackOnRollOff</code> | <i>If true, will automatically go back to the prior state on roll off of the rollover event.</i> |
| id | <code>integer •id</code> | <i>The unique ID of the GotoStateBehavior.</i> |
| index | <code>integer •index</code> | <i>The index of the GotoStateBehavior within its containing object.</i> |
| label | <code>string label</code> | <i>A property that can be set to any string.</i> |
| name | <code>string •name</code> | <i>The name of the GotoStateBehavior.</i> |
| parent | <code>{Button} •parent</code> | <i>The parent of the GotoStateBehavior (a Button).</i> |
| stateName | <code>string stateName</code> | <i>The name of the state in the associated multi-state object.</i> |

GotoStateBehaviors

Object available in ID 7

| GOTOSTATEBEHAVIORS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|---|
| add | <code>{GotoStateBehavior} add()</code> | <i>Creates a new GotoStateBehavior. / Return value: The new GotoStateBehavior.</i> |
| anyItem | <code>IDispatch anyItem()</code> | <i>Returns any object in the collection. / Return value: The object.</i> |
| firstItem | <code>IDispatch firstItem()</code> | <i>Returns the first object in the collection. / Return value: The first object.</i> |
| itemByID | <code>IDispatch itemByID(integer Id)</code> | <i>Returns the object with the specified ID. / Return value: The object. / ID: The ID.</i> |
| itemByRange | <code>{Objects} itemByRange(variant From, variant To)</code> | <i>Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String.</i> |
| lastItem | <code>IDispatch lastItem()</code> | <i>Returns the last object in the collection. / Return value: The last object.</i> |
| middleItem | <code>IDispatch middleItem()</code> | <i>Returns the middle object in the collection. / Return value: The middle object.</i> |
| nextItem | <code>IDispatch nextItem(IDispatch Obj)</code> | <i>Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object.</i> |
| previousItem | <code>IDispatch previousItem(IDispatch Obj)</code> | <i>Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object.</i> |
| count | <code>integer •count</code> | <i>Displays the number of elements in the GotoStateBehavior.</i> |
| item | <code>{GotoStateBehavior} •item(variant Index)</code> | <i>Returns the object with the specified index or name. Type: Long Integer or String.</i> |

GraphicEvents

Object available in ID 7

| GRAPHICEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|---|
| afterPlace | <code>const afterPlace(afterPlace)</code> | Dispatched after a Graphic is placed. This event bubbles. This event is not cancelable. |
| beforePlace | <code>const beforePlace(beforePlace)</code> | Dispatched before a Graphic is placed. This event bubbles. This event is cancelable. |

GraphicLineEvents

Object available in ID 7

| GRAPHICLINEEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|--|---|
| afterPlace | <code>const afterPlace(afterPlace)</code> | Dispatched after a GraphicLine is placed. This event bubbles. This event is not cancelable. |
| beforePlace | <code>const beforePlace(beforePlace)</code> | Dispatched before a GraphicLine is placed. This event bubbles. This event is cancelable. |

GridDataInformation

Object available in ID 7

| GRIDDATAINFORMATION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|---|---|
| appliedFont | <code>variant appliedFont</code> | The font applied to the GridDataInformation, specified as either a font object or the name of font family. Type: Font or String. |
| characterAki | <code>double characterAki</code> | The amount of white space between characters. |
| characterAlignment | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| characterCountLocation | <code>#CharacterCountLocation characterCountLocation</code> | The character count location. Note: Valid when show character count is true. |
| characterCountSize | <code>double characterCountSize</code> | The character size for the character count display. |
| fontStyle | <code>string fontStyle</code> | The name of the font style. |
| gridAlignment | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| gridView | <code>#GridViewSettings gridView</code> | The grid view setting. |
| horizontalScale | <code>double horizontalScale</code> | The horizontal scaling applied to the GridDataInformation. |
| lineAki | <code>double lineAki</code> | The amount of white space between lines. |
| lineAlignment | <code>#LineAlignment lineAlignment</code> | The line justification. |
| parent | <code>variant •parent</code> | The parent of the GridDataInformation (a Story, XmlStory, Page, NamedGrid or TextFrame). Type: Story, XmlStory, Page, NamedGrid or TextFrame. |
| pointSize | <code>variant pointSize</code> | The text size. Type: Unit (Double or String). |
| verticalScale | <code>double verticalScale</code> | The vertical scaling applied to the GridDataInformation. |

GridPrintingPreference

Object available in ID 7

| GRIDPRINTINGPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|--|
| <code>frameGridPrinting</code> | <code>boolean frameGridPrinting</code> | If true, displays frame (story) grids in the printed or exported document. |
| <code>frameGridStrokeWeight</code> | <code>double frameGridStrokeWeight</code> | The stroke weight (in points) of the frame grid. |
| <code>layoutGridPrinting</code> | <code>boolean layoutGridPrinting</code> | If true, displays layout grids in the printed or exported document. |
| <code>layoutGridStrokeWeight</code> | <code>double layoutGridStrokeWeight</code> | The stroke weight (in points) of the layout grid. |
| <code>pageItemPrinting</code> | <code>boolean pageItemPrinting</code> | If true, displays page items other than text, frame grids, and layout grids in the printed or exported document. For information on printing and exporting text and grids, see text printing, frame grid printing, and layout grid printing. |
| <code>parent</code> | <code>{Application} *parent</code> | The parent of the GridPrintingPreference (a Application). |
| <code>textPrinting</code> | <code>boolean textPrinting</code> | If true, displays text in the printed or exported document. |

GroupEvents

Object available in ID 7

| GROUPEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|--|---|
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | Dispatched after a Group is placed. This event bubbles. This event is not cancelable. |
| <code>beforePlace</code> | <code>const beforePlace(beforePlace)</code> | Dispatched before a Group is placed. This event bubbles. This event is cancelable. |

GuideEvents

Object available in ID 7

| GUIDEEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------|--|---|
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | Dispatched after a Guide is placed. This event bubbles. This event is not cancelable. |

IdleEvent

Object available in ID 7

| IDLEEVENT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|-------------------------------------|--|
| <code>preventDefault</code> | <code>void preventDefault()</code> | Cancels the default behavior of the event on its target. |
| <code>stopPropagation</code> | <code>void stopPropagation()</code> | Stops propagation of the event beyond the current target. |
| <code>bubbles</code> | <code>boolean *bubbles</code> | If true, the event supports the bubbling phase of propagation. |

| IDLEEVENT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|--|---|
| <code>cancelable</code> | <code>boolean *cancelable</code> | If true, the default behavior of the event on its target can be canceled. |
| <code>currentTarget</code> | <code>IDispatch *currentTarget</code> | The current propagation target of the event. |
| <code>defaultPrevented</code> | <code>boolean *defaultPrevented</code> | If true, the default behavior of the event on its target has been canceled. |
| <code>eventPhase</code> | <code>#EventPhases *eventPhase</code> | The current propagation phase of the event. |
| <code>eventType</code> | <code>string *eventType</code> | The name of the event. |
| <code>id</code> | <code>integer *id</code> | The unique ID of the IdleEvent. |
| <code>index</code> | <code>integer *index</code> | The index of the IdleEvent within its containing object. |
| <code>parent</code> | <code>{IdleTask} *parent</code> | The parent of the IdleEvent (a IdleTask). |
| <code>propagationStopped</code> | <code>boolean *propagationStopped</code> | If true, propagation of the event beyond the current target has been stopped. |
| <code>target</code> | <code>IDispatch *target</code> | The target of the event. |
| <code>timeAllocated</code> | <code>integer *timeAllocated</code> | Amount of time allocated to this task at the time the event was dispatched. |
| <code>timeStamp</code> | <code>date *timeStamp</code> | The time the event was initialized. |

IdleEventEvents

Object available in ID 7

| IDLEEVENTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|------------------------------------|---|
| <code>onIdle</code> | <code>const onIdle(onIdle)</code> | Dispatched at idle time for this IdleEvent. This event does not bubble. This event is not cancelable. |

IdleTask

Object available in ID 7

| IDLETASK CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|---|--|
| <code>delete</code> | <code>void delete()</code> | Deletes the IdleTask. |
| <code>extractLabel</code> | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| <code>insertLabel</code> | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| <code>id</code> | <code>integer *id</code> | The unique ID of the IdleTask. |
| <code>index</code> | <code>integer *index</code> | The index of the IdleTask within its containing object. |
| <code>label</code> | <code>string label</code> | A property that can be set to any string. |
| <code>name</code> | <code>string name</code> | The name of the IdleTask. |
| <code>parent</code> | <code>{Application} *parent</code> | The parent of the IdleTask (a Application). |

| IDLETASK CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------|----------------------------|--|
| sleep | <code>integer sleep</code> | <i>Amount of time to sleep (in milliseconds) before calling this task again. Setting this to zero will cause task to be deleted.</i> |

IdleTaskEvents

Object available in ID 7

| IDLETASKEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|------------------------------------|---|
| onIdle | <code>const onIdle(onIdle)</code> | <i>Dispatched at idle time for this IdleTask. This event does not bubble. This event is not cancelable.</i> |

IdleTasks

Object available in ID 7

| IDLETASKS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------|--|---|
| add | <code>{IdleTask} add()</code> | <i>Creates a new IdleTask. / Return value: The new IdleTask.</i> |
| anyItem | <code>IDispatch anyItem()</code> | <i>Returns any object in the collection. / Return value: The object.</i> |
| firstItem | <code>IDispatch firstItem()</code> | <i>Returns the first object in the collection. / Return value: The first object.</i> |
| itemByID | <code>IDispatch itemByID(integer Id)</code> | <i>Returns the object with the specified ID. / Return value: The object. / ID: The ID.</i> |
| itemByRange | <code>{Objects} itemByRange(variant From, variant To)</code> | <i>Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String.</i> |
| lastItem | <code>IDispatch lastItem()</code> | <i>Returns the last object in the collection. / Return value: The last object.</i> |
| middleItem | <code>IDispatch middleItem()</code> | <i>Returns the middle object in the collection. / Return value: The middle object.</i> |
| nextItem | <code>IDispatch nextItem(IDispatch Obj)</code> | <i>Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object.</i> |
| previousItem | <code>IDispatch previousItem(IDispatch Obj)</code> | <i>Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object.</i> |
| count | <code>integer *count</code> | <i>Displays the number of elements in the IdleTask.</i> |
| item | <code>{IdleTask} *item(variant Index)</code> | <i>Returns the object with the specified index or name. Type: Long Integer or String.</i> |

ImageEvents

Object available in ID 7

| IMAGEEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------|--|---|
| afterPlace | <code>const afterPlace(afterPlace)</code> | Dispatched after a Image is placed. This event bubbles. This event is not cancelable. |
| beforePlace | <code>const beforePlace(beforePlace)</code> | Dispatched before a Image is placed. This event bubbles. This event is cancelable. |

ImportedPageEvents

Object available in ID 7

| IMPORTEDPAGEEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|--|--|
| afterPlace | <code>const afterPlace(afterPlace)</code> | Dispatched after a ImportedPage is placed. This event bubbles. This event is not cancelable. |
| beforePlace | <code>const beforePlace(beforePlace)</code> | Dispatched before a ImportedPage is placed. This event bubbles. This event is cancelable. |

ImportExportEventEvents

Object available in ID 7

| IMPORTEXPORTEVENTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------------|--|---|
| afterExport | <code>const afterExport(afterExport)</code> | Dispatched after a ImportExportEvent is exported. This event bubbles. This event is not cancelable. |
| afterImport | <code>const afterImport(afterImport)</code> | Dispatched after importing a file into a ImportExportEvent. This event bubbles. This event is not cancelable. |
| beforeExport | <code>const beforeExport(beforeExport)</code> | Dispatched before a ImportExportEvent is exported. This event bubbles. This event is cancelable. |
| beforeImport | <code>const beforeImport(beforeImport)</code> | Dispatched before importing a file into a ImportExportEvent. This event bubbles. This event is cancelable. |
| failedExport | <code>const failedExport(failedExport)</code> | Dispatched after a ImportExportEvent export is canceled or fails. This event bubbles. This event is not cancelable. |

InteractivePDFExportPreference

Object available in ID 7

| INTERACTIVEPDFEXPORTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|--|
| exportLayers | <code>boolean exportLayers</code> | If true, saves each layer as an Acrobat layer within the PDF document. |
| exportReaderSpreads | <code>boolean exportReaderSpreads</code> | If true, each spread in the exported document is combined into a single page that has spread's original width. |

| INTERACTIVEPDFEXPORTPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|---|
| <code>flipPages</code> | <code>boolean flipPages</code> | Automatically flip pages in the exported PDF. |
| <code>flipPagesSpeed</code> | <code>integer flipPagesSpeed</code> | The speed that the pages flip. Type: Long Integer (0 - 32767). |
| <code>generateThumbnails</code> | <code>boolean generateThumbnails</code> | If true, generates thumbnail images for each page or spread. |
| <code>includeStructure</code> | <code>boolean includeStructure</code> | If true, creates a tagged PDF file. Note: If acrobat compatibility is acrobat 6 or higher, tags are visible only when the PDF is opened in Acrobat 6 or higher. |
| <code>interactivePDFInteractiveElementsOption</code> | <code>#InteractivePDFInteractiveElementsOptions interactivePDFInteractiveElementsOption</code> | How to draw interactive elements. |
| <code>openInFullScreen</code> | <code>boolean openInFullScreen</code> | Open PDF in full screen mode. |
| <code>pageRange</code> | <code>variant pageRange</code> | The pages to print, specified either as an enumeration or a string. To specify a range, separate page numbers in the string with a hyphen (-). To specify separate pages, separate page numbers in the string with a comma (,). Type: idPageRange enumerator or String. |
| <code>pageTransitionOverride</code> | <code>#PageTransitionOverrideOptions pageTransitionOverride</code> | The name of the page transition to use for all pages. |
| <code>parent</code> | <code>{Application} *parent</code> | The parent of the InteractivePDFExportPreference (a Application). |
| <code>pdfJPEGQuality</code> | <code>#PDFJPEGQualityOptions pdfJPEGQuality</code> | The PDF JPEG quality options. |
| <code>pdfMagnification</code> | <code>#PdfMagnificationOptions pdfMagnification</code> | The export PDF magnification options. |
| <code>pdfPageLayout</code> | <code>#PageLayoutOptions pdfPageLayout</code> | The export PDF page layout. |
| <code>pdfRasterCompression</code> | <code>#PDFRasterCompressionOptions pdfRasterCompression</code> | The PDF raster compression options. |
| <code>rasterResolution</code> | <code>variant rasterResolution</code> | The raster resolution. Type: idRasterResolutionOptions enumerator or Long Integer (72 - 300). |
| <code>viewPDF</code> | <code>boolean viewPDF</code> | If true, automatically opens the PDF file after exporting. |

KinsokuTable

Object available in ID 7

| KINSOKUTABLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|---|--|
| <code>delete</code> | <code>void delete()</code> | Deletes the KinsokuTable. |
| <code>extractLabel</code> | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| <code>insertLabel</code> | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| <code>cantBeginLineChars</code> | <code>string cantBeginLineChars</code> | The characters in the kinsoku set that cannot begin lines. |
| <code>cantBeSeparatedChars</code> | <code>string cantBeSeparatedChars</code> | The characters in the kinsoku set that cannot be separated. |
| <code>cantEndLineChars</code> | <code>string cantEndLineChars</code> | That characters in the kinsoku set that cannot end lines. |
| <code>hangingPunctuationChars</code> | <code>string hangingPunctuationChars</code> | The hanging punctuation characters in the kinsoku set. |
| <code>id</code> | <code>integer *id</code> | The unique ID of the KinsokuTable. |
| <code>index</code> | <code>integer *index</code> | The index of the KinsokuTable within its containing object. |

| KINSOKUTABLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|------------------------------|--|
| <code>label</code> | <code>string label</code> | A property that can be set to any string. |
| <code>name</code> | <code>string name</code> | The name of the KinsokuTable. |
| <code>parent</code> | <code>variant •parent</code> | The parent of the KinsokuTable (a Document or Application). Type: Document or Application. |

KinsokuTables

Object available in ID 7

| KINSOKUTABLES CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|--|
| <code>add</code> | <code>{KinsokuTable} add(string Name)</code> | Creates a new kinsoku table. / Return value: The new kinsoku table. / Name: The name of the new kinsoku table. |
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | Returns any object in the collection. / Return value: The object. |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | Returns the first object in the collection. / Return value: The first object. |
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | Returns the object with the specified ID. / Return value: The object. / ID: The ID. |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| <code>previousItem</code> | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| <code>count</code> | <code>integer •count</code> | Displays the number of elements in the KinsokuTable. |
| <code>item</code> | <code>{KinsokuTable} •item(variant Index)</code> | Returns the object with the specified index or name. Type: Long Integer or String. |

LayoutGridDataInformation

Object available in ID 7

| LAYOUTGRIDDATAINFORMATION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|----------------------------------|--|
| <code>appliedFont</code> | <code>variant appliedFont</code> | The font applied to the LayoutGridDataInformation, specified as either a font object or the name of font family. Type: Font or String. |
| <code>characterAki</code> | <code>double characterAki</code> | The amount of white space between characters. |
| <code>fontStyle</code> | <code>string fontStyle</code> | The name of the font style. |

| LAYOUTGRIDDATAINFORMATION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|-------------------------------------|--|
| <code>horizontalScale</code> | <code>double horizontalScale</code> | <i>The horizontal scaling applied to the LayoutGridDataInformation.</i> |
| <code>lineAki</code> | <code>double lineAki</code> | <i>The amount of white space between lines.</i> |
| <code>parent</code> | <code>variant *parent</code> | <i>The parent of the LayoutGridDataInformation (a Application or Document). Type: Application or Document.</i> |
| <code>pointSize</code> | <code>variant pointSize</code> | <i>The text size. Type: Unit (Double or String).</i> |
| <code>verticalScale</code> | <code>double verticalScale</code> | <i>The vertical scaling applied to the LayoutGridDataInformation.</i> |

LayoutWindowEvents

Object available in ID 7

| LAYOUTWINDOWEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|---|
| <code>afterActivate</code> | <code>const afterActivate(afterActivate)</code> | <i>Dispatched after the LayoutWindow becomes active. This event bubbles. This event is not cancelable.</i> |
| <code>afterAttributeChanged</code> | <code>const afterAttributeChanged(afterAttributeChanged)</code> | <i>Dispatched when the value of a property changes on this LayoutWindow. This event bubbles. This event is not cancelable.</i> |
| <code>afterClose</code> | <code>const afterClose(afterClose)</code> | <i>Dispatched when a LayoutWindow is closing. Since the close has been committed, it can no longer be canceled. This event bubbles. This event is not cancelable.</i> |
| <code>afterOpen</code> | <code>const afterOpen(afterOpen)</code> | <i>Dispatched after a LayoutWindow is opened. This event bubbles. This event is not cancelable.</i> |
| <code>afterSelectionAttributeChanged</code> | <code>const afterSelectionAttributeChanged(afterSelectionAttributeChanged)</code> | <i>Dispatched after an attribute on the active selection changes. This event bubbles. This event is not cancelable.</i> |
| <code>afterSelectionChanged</code> | <code>const afterSelectionChanged(afterSelectionChanged)</code> | <i>Dispatched after the active selection changes. This event bubbles. This event is not cancelable.</i> |
| <code>beforeClose</code> | <code>const beforeClose(beforeClose)</code> | <i>Dispatched before a LayoutWindow is closed. This event bubbles. This event is cancelable.</i> |
| <code>beforeDeactivate</code> | <code>const beforeDeactivate(beforeDeactivate)</code> | <i>Dispatched before the LayoutWindow becomes inactive. This event bubbles. This event is not cancelable.</i> |

LinkEvents

Object available in ID 7

| LINEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|--|
| <code>afterAttributeChanged</code> | <code>const afterAttributeChanged(afterAttributeChanged)</code> | <i>Dispatched when the value of a property changes on this Link. This event bubbles. This event is not cancelable.</i> |
| <code>afterDelete</code> | <code>const afterDelete(afterDelete)</code> | <i>Dispatched after a Link is deleted. This event bubbles. This event is not cancelable.</i> |
| <code>afterEmbed</code> | <code>const afterEmbed(afterEmbed)</code> | <i>Dispatched after a Link is embedded. This event bubbles. This event is not cancelable.</i> |

| LINKEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------|--|---|
| afterMove | <code>const afterMove(afterMove)</code> | <i>Dispatched after a Link is relocated from one object to another. This event bubbles. This event is not cancelable.</i> |
| afterNew | <code>const afterNew(afterNew)</code> | <i>Dispatched after a Link is created. This event bubbles. This event is not cancelable.</i> |
| afterUnembed | <code>const afterUnembed(afterUnembed)</code> | <i>Dispatched after a Link is unembedded. This event bubbles. This event is not cancelable.</i> |
| afterUpdate | <code>const afterUpdate(afterUpdate)</code> | <i>Dispatched after a Link is updated. This event bubbles. This event is not cancelable.</i> |
| beforeDelete | <code>const beforeDelete(beforeDelete)</code> | <i>Dispatched before a Link is deleted. This event bubbles. This event is cancelable.</i> |
| beforeEmbed | <code>const beforeEmbed(beforeEmbed)</code> | <i>Dispatched before a Link is embedded. This event bubbles. This event is cancelable.</i> |
| beforeMove | <code>const beforeMove(beforeMove)</code> | <i>Dispatched before a Link is relocated from one object to another. This event bubbles. This event is cancelable.</i> |
| beforeUnembed | <code>const beforeUnembed(beforeUnembed)</code> | <i>Dispatched before a Link is unembedded. This event bubbles. This event is cancelable.</i> |
| beforeUpdate | <code>const beforeUpdate(beforeUpdate)</code> | <i>Dispatched before a Link is updated. This event bubbles. This event is cancelable.</i> |

MediaItem

Object available in ID 7

| MEDIAITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|---|
| applyObjectStyle | <code>void applyObjectStyle(ObjectStyle Using, boolean ClearingOverrides, boolean ClearingOverridesThroughRootObjectStyle)</code> | <i>Applies the specified object style. / Using: The object style to apply. / ClearingOverrides: If true, clears the MediaItem's existing attributes before applying the style. / ClearingOverridesThroughRootObjectStyle: If true, clears attributes and formatting applied to the MediaItem that are not defined in the object style.</i> |
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| autoTag | <code>void autoTag()</code> | <i>Tag the object or the parent story using default tags defined in XML preference.</i> |
| changeObject | <code>{Objects} changeObject(variant ReverseOrder)</code> | <i>Finds objects that match the find what value and replace the objects with the change to value. / Return value: The replaced object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean.</i> |
| clearObjectStyleOverrides | <code>void clearObjectStyleOverrides()</code> | <i>Clear overrides for object style.</i> |
| clearTransformations | <code>void clearTransformations()</code> | <i>Clears transformations from the MediaItem. Transformations include rotation, scaling, flipping, fitting, and shearing.</i> |

| MEDIAITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------|--|--|
| convertShape | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | Converts the MediaItem to a different shape. / Given: The MediaItem's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0) . Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points). |
| delete | <code>void delete()</code> | Deletes the MediaItem. |
| detach | <code>void detach()</code> | Detaches an overridden master page item from the master page. |
| duplicate | <code>{PageItem} duplicate(variant To, variant By)</code> | Duplicates the MediaItem at the specified location or offset. / Return value: The duplicated MediaItem. / To: The location of the new MediaItem, specified in coordinates in the format [x, y]. Type: Array of 2 Units (Doubles or Strings), Spread, Page or Layer. / By: Amount by which to offset the new MediaItem from the original MediaItem's position. Type: Array of 2 Units (Doubles or Strings). |
| export | <code>void export(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | Exports the object(s) to a file. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| extractLabel | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| findObject | <code>{Objects} findObject(variant ReverseOrder)</code> | Finds objects that match the find what value. / Return value: The found object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| fit | <code>void fit(#FitOptions Given)</code> | Applies the specified fit option to content in a frame. / Given: The fit option to use. |
| flipItem | <code>void flipItem(#Flip Given, variant Around)</code> | Flips the MediaItem. / Given: The axis around which to flip the MediaItem. / Around: The point around which to flip the MediaItem. Type: Array of 2 Units (Doubles or Strings) or idAnchorPoint enumerator. |
| insertLabel | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| markup | <code>void markup(XMLElement Using)</code> | Associates the page item with the specified XML element while preserving existing content. / Using: The XML element. |
| move | <code>void move(variant To, variant By)</code> | Moves the MediaItem to a new location. Note: Either the 'to' or 'by' parameter is required; if both parameters are defined, only the to value is used. / To: The new location of the MediaItem,in the format (x, y). Type: Array of 2 Units (Doubles or Strings), Spread, Page or Layer. / By: The amount (in measurement units) to move the MediaItem relative to its current position, in the format (x, y). Type: Array of 2 Units (Doubles or Strings). |
| override | <code>variant override(Page DestinationPage)</code> | Overrides a master page item and places the item on the document page as a new object. / Return value: The new object. Type: PageItem, Guide, Graphic, Movie or Sound. / DestinationPage: The document page that contains the master page item to override. |
| place | <code>variant place(string FileName, boolean ShowingOptions)</code> | Places the file. / Return value: The placed file. Type: Array of Pageitems, Graphics, Movies, Sounds, Stories or Texts. / FileName: The file to place. / ShowingOptions: Whether to display the import options dialog. |

| MEDIAITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|---|
| <code>placeXML</code> | <code>void placeXML(XMLElement Using)</code> | Places XML content into the specified object. / <code>Using</code> : The XML element whose content you want to place. |
| <code>redefineScaling</code> | <code>void redefineScaling(variant To)</code> | Apply an item's scaling to its content if possible. / <code>To</code> : The scale factors to be left on the item. The default is {1.0, 1.0}. Type: Array of 2 Doubles. |
| <code>reframe</code> | <code>void reframe(variant In, variant OpposingCorners)</code> | Move the bounding box of the page item. / <code>In</code> : The bounding box to resize. Type: <code>idCoordinateSpaces</code> enumerator or Ordered array containing <code>CoordinateSpace:idCoordinateSpaces</code> enumerator, <code>BoundsKind:idBoundingBoxLimits</code> enumerator. / <code>OpposingCorners</code> : Opposing corners of new bounding box in the given coordinate space. Type: Array of 2 Arrays of 2 Doubles. |
| <code>removeOverride</code> | <code>void removeOverride()</code> | Removes the override from a previously overridden master page item. |
| <code>resize</code> | <code>void resize(variant In, variant From, #ResizeMethods By, variant Values, boolean ResizeIndividually, boolean ConsideringRulerUnits)</code> | Resize the page item. / <code>In</code> : The bounding box to resize. Type: <code>idCoordinateSpaces</code> enumerator, <code>idBoundingBoxLimits</code> enumerator or Ordered array containing <code>CoordinateSpace:idCoordinateSpaces</code> enumerator, <code>BoundsKind:idBoundingBoxLimits</code> enumerator. / <code>From</code> : The transform origin. Legal specifications: relative to bounding box: <code>anchor</code> <code>{anchor}</code> <code>{x,y}</code> , <code>bounds kind [, coordinate space]</code> ; relative to coordinate space: <code>{x,y}</code> <code>{x,y [, coordinate space]}</code> ; relative to layout window ruler: <code>{x,y}</code> , <code>page index</code> <code>bounds kind</code> . Type: Array of 2 Doubles, <code>idAnchorPoint</code> enumerator or Array of Arrays of 2 Doubles, <code>idCoordinateSpaces</code> enumerators, <code>idAnchorPoint</code> enumerators, <code>idBoundingBoxLimits</code> enumerators or Long Integers. / <code>By</code> : How the current dimensions are affected by the given values. / <code>Values</code> : The width and height values. Legal dimensions specifications: <code>{x, y [, coordinate space]}</code> , <code>{x, resize constraint [, coordinate space]}</code> , or <code>{resize constraint, y [, coordinate space]}</code> ; where x and y are real numbers and coordinate space is used to determine _only_ the unit of length for x and y; coordinate space is ignored for the 'current dimensions times' resize method). Type: Array of Doubles, <code>idResizeConstraints</code> enumerators or <code>idCoordinateSpaces</code> enumerators. / <code>ResizeIndividually</code> : If false and multiple page items are targeted, the new dimensions are attained only by moving the individual items rather than resizing them. / <code>ConsideringRulerUnits</code> : If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| <code>resolve</code> | <code>variant resolve(variant Location, #CoordinateSpaces In, boolean ConsideringRulerUnits)</code> | Get the coordinates of the given location in the specified coordinate system. / Return value: The resolved location for each of the targeted page items. Type: Array of 2 Doubles or Array of Arrays of 2 Doubles. / <code>Location</code> : The location requested. Type: Array of 2 Doubles, <code>idAnchorPoint</code> enumerator or Array of Arrays of 2 Doubles, <code>idCoordinateSpaces</code> enumerators, <code>idAnchorPoint</code> enumerators, <code>idBoundingBoxLimits</code> enumerators or Long Integers. / <code>In</code> : The coordinate space to use. / <code>ConsideringRulerUnits</code> : If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| <code>select</code> | <code>void select(#SelectionOptions ExistingSelection)</code> | Selects the object. / <code>ExistingSelection</code> : The selection status of the Mediaitem in relation to previously selected objects. |
| <code>store</code> | <code>{Asset} store(Library Using)</code> | Stores the object in the specified library. / Return value: The new library asset(s). / <code>Using</code> : The library in which to store the object. |

| MEDIAITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| transform | <code>void transform(#CoordinateSpaces In, variant From, variant WithMatrix, variant ReplacingCurrent, boolean ConsideringRulerUnits)</code> | Transform the page item. / In: The coordinate space to use. / From: The temporary origin during the transformation. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / WithMatrix: Transform matrix. Type: Array of 6 Doubles or TransformationMatrix. / ReplacingCurrent: Transform components to consider; providing this optional parameter causes the target's existing transform components to be replaced with new values. Without this parameter, the given matrix is concatenated onto the target's existing transform combining the effect of the two. Type: idMatrixContent enumerator, Array of idMatrixContent enumerators or Long Integer. / ConsideringRulerUnits: If true then a ruler based origin is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| transformAgain | <code>variant transformAgain()</code> | Transforms the MediaItem using the last transformation performed on any object. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformAgainIndividually | <code>variant transformAgainIndividually()</code> | Transforms the MediaItem using the last transformation performed on any MediaItem. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformSequenceAgain | <code>variant transformSequenceAgain()</code> | Transforms the MediaItem using the last sequence of transform operations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformSequenceAgainIndividually | <code>variant transformSequenceAgainIndividually()</code> | Transforms the MediaItem using the last sequence of transformations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformValuesOf | <code>{Objects} transformValuesOf(#CoordinateSpaces In)</code> | Get the transformation values of the page item. / Return value: A list of the transformation matrices. / In: The coordinate space to use. |
| absoluteFlip | <code>#Flip absoluteFlip</code> | Indicates whether the MediaItem has been flipped independently of its parent object and, if yes, the direction in which the MediaItem was flipped. |
| absoluteHorizontalScale | <code>double absoluteHorizontalScale</code> | The horizontal scale of the MediaItem relative to its containing object. |
| absoluteRotationAngle | <code>double absoluteRotationAngle</code> | The rotation angle of the MediaItem relative to its containing object. (Range: -360 to 360). |
| absoluteShearAngle | <code>double absoluteShearAngle</code> | The skewing angle of the MediaItem relative to its containing object. (Range: -360 to 360). |
| absoluteVerticalScale | <code>double absoluteVerticalScale</code> | The vertical scale of the MediaItem relative to its containing object. |
| allGraphics | <code>{Objects} *allGraphics</code> | Lists all graphics contained by the MediaItem. |
| allowOverrides | <code>boolean allowOverrides</code> | If true, the master page item can be overridden. |
| allPageItems | <code>{Objects} *allPageItems</code> | Lists all page items contained by the MediaItem. |
| animationSettings | <code>{AnimationSetting} *animationSettings</code> | The page item animation settings. |
| appliedObjectStyle | <code>{ObjectStyle} appliedObjectStyle</code> | The object style applied to the MediaItem. |
| associatedXMLElement | <code>{XMLElement} *associatedXMLElement</code> | The XML element associated with the MediaItem. |
| bottomLeftCornerOption | <code>#CornerOptions bottomLeftCornerOption</code> | The shape to apply to the bottom left corner of rectangular shapes. |

| MEDIAITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|---|
| bottomLeftCornerRadius | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| bottomRightCornerOption | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| bottomRightCornerRadius | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| contentTransparencySettings | <code>{ContentTransparencySetting} *contentTransparencySettings</code> | Transparency settings for the content of the Mediaitem. |
| endCap | <code>#EndCap endCap</code> | The end shape of an open path. |
| endJoin | <code>#EndJoin endJoin</code> | The corner join applied to the Mediaitem. |
| fillColor | <code>variant fillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of the Mediaitem. . Type: Swatch or String. |
| fillTint | <code>double fillTint</code> | The percent of tint to use in the Mediaitem's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| fillTransparencySettings | <code>{FillTransparencySetting} *fillTransparencySettings</code> | Transparency settings for the fill applied to the Mediaitem. |
| flip | <code>#Flip flip</code> | The direction in which to flip the printed image. |
| gapColor | <code>{Swatch} gapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type. |
| gapTint | <code>double gapTint</code> | The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| geometricBounds | <code>variant geometricBounds</code> | The bounds of the Mediaitem excluding the stroke width, in the format [y1, x1, y2, x2], which give the coordinates of the top-left and bottom-right corners of the bounding box. Type: Array of 4 Units (Doubles or Strings). |
| gradientFillAngle | <code>double gradientFillAngle</code> | The angle of a linear gradient applied to the fill of the Mediaitem. (Range: -180 to 180). |
| gradientFillLength | <code>variant gradientFillLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the Mediaitem. Type: Unit (Double or String). |
| gradientFillStart | <code>variant gradientFillStart</code> | The starting point (in page coordinates) of a gradient applied to the fill of the Mediaitem, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| gradientStrokeAngle | <code>double gradientStrokeAngle</code> | The angle of a linear gradient applied to the stroke of the Mediaitem. (Range: -180 to 180). |
| gradientStrokeLength | <code>variant gradientStrokeLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the Mediaitem. Type: Unit (Double or String). |
| gradientStrokeStart | <code>variant gradientStrokeStart</code> | The starting point (in page coordinates) of a gradient applied to the stroke of the Mediaitem, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| horizontalScale | <code>double horizontalScale</code> | The horizontal scaling applied to the Mediaitem. |
| id | <code>integer *id</code> | The unique ID of the Mediaitem. |
| index | <code>integer *index</code> | The index of the Mediaitem within its containing object. |
| itemLayer | <code>{Layer} itemLayer</code> | The layer that the Mediaitem is on. |
| label | <code>string label</code> | A property that can be set to any string. |
| leftLineEnd | <code>#ArrowHead leftLineEnd</code> | The arrowhead applied to the start of the path. |

| MEDIAITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|---|
| localDisplaySetting | <code>#DisplaySettingOptions localDisplaySetting</code> | <i>Display performance options for the MediaItem.</i> |
| locked | <code>boolean locked</code> | <i>If true, the MediaItem is locked.</i> |
| miterLimit | <code>double miterLimit</code> | <i>The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500).</i> |
| name | <code>string name</code> | <i>The name of the MediaItem.</i> |
| nonprinting | <code>boolean nonprinting</code> | <i>If true, the MediaItem does not print.</i> |
| overprintFill | <code>boolean overprintFill</code> | <i>If true, the MediaItem's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors.</i> |
| overprintGap | <code>boolean overprintGap</code> | <i>If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors.</i> |
| overprintStroke | <code>boolean overprintStroke</code> | <i>If true, the MediaItem's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors.</i> |
| overridden | <code>boolean *overridden</code> | <i>If true, the object originated on a master spread and was overridden. If false, the object either originated on a master spread and was not overridden, or the object did not originate on a master page.</i> |
| overriddenMasterPageItem | <code>variant *overriddenMasterPageItem</code> | <i>An object that originated on a master page and has been overridden. Type: PageItem, Guide, Graphic, Movie or Sound.</i> |
| parent | <code>variant *parent</code> | <i>The parent of the MediaItem (a Polygon, GraphicLine, Rectangle, Oval or SplineItem). Type: Polygon, GraphicLine, Rectangle, Oval or SplineItem.</i> |
| parentPage | <code>{Page} *parentPage</code> | <i>The page on which this page item appears.</i> |
| paths | <code>{Paths} *paths</code> | <i>A collection of paths.</i> |
| preferences | <code>{Preferences} *preferences</code> | <i>A collection of preferences objects.</i> |
| rightLineEnd | <code>#ArrowHead rightLineEnd</code> | <i>The arrowhead applied to the end of the path.</i> |
| rotationAngle | <code>double rotationAngle</code> | <i>The rotation angle of the MediaItem. (Range: -360 to 360).</i> |
| shearAngle | <code>double shearAngle</code> | <i>The skewing angle applied to the MediaItem. (Range: -360 to 360).</i> |
| strokeAlignment | <code>#StrokeAlignment strokeAlignment</code> | <i>The stroke alignment applied to the MediaItem.</i> |
| strokeColor | <code>variant strokeColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the MediaItem. Type: Swatch or String.</i> |
| strokeCornerAdjustment | <code>#StrokeCornerAdjustment strokeCornerAdjustment</code> | <i>The corner adjustment applied to the MediaItem.</i> |
| strokeDashAndGap | <code>variant strokeDashAndGap</code> | <i>The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings).</i> |
| strokeTint | <code>double strokeTint</code> | <i>The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| strokeTransparencySettings | <code>{StrokeTransparencySetting} *strokeTransparencySettings</code> | <i>Transparency settings for the stroke.</i> |
| strokeType | <code>variant strokeType</code> | <i>The name of the stroke style to apply. Type: StrokeStyle or String.</i> |
| strokeWeight | <code>variant strokeWeight</code> | <i>The weight (in points) to apply to the MediaItem's stroke. Type: Unit (Double or String).</i> |
| textWrapPreferences | <code>{TextWrapPreference} *textWrapPreferences</code> | <i>The text wrap preference properties that define the default formatting for wrapping text around objects.</i> |

| MEDIAITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|--|
| <code>timingSettings</code> | <code>{TimingSetting} *timingSettings</code> | <i>The object timing settings.</i> |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>transparencySettings</code> | <code>{TransparencySetting} *transparencySettings</code> | <i>Transparency settings.</i> |
| <code>verticalScale</code> | <code>double verticalScale</code> | <i>The vertical scaling applied to the MediaItem.</i> |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the MediaItem is visible.</i> |
| <code>visibleBounds</code> | <code>variant visibleBounds</code> | <i>The bounds of the MediaItem including the stroke width, in the format [y1, x1, y2, x2], which give the coordinates of the top-left and bottom-right corners of the bounding box. Type: Array of 4 Units (Doubles or Strings).</i> |

MediaItemEvents

Object available in ID 7

| MEDIAITEMEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|---|
| <code>afterAttributeChanged</code> | <code>const afterAttributeChanged(afterAttributeChanged)</code> | <i>Dispatched when the value of a property changes on this MediaItem. This event bubbles. This event is not cancelable.</i> |
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a MediaItem is placed. This event bubbles. This event is not cancelable.</i> |
| <code>beforePlace</code> | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a MediaItem is placed. This event bubbles. This event is cancelable.</i> |

MediaItems

Object available in ID 7

| MEDIAITEMS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|--|---|
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | <i>Returns any object in the collection. / Return value: The object.</i> |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | <i>Returns the first object in the collection. / Return value: The first object.</i> |
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | <i>Returns the object with the specified ID. / Return value: The object. / ID: The ID.</i> |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | <i>Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String.</i> |

| MEDIAITEMS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|--|
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| <code>previousItem</code> | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| <code>count</code> | <code>integer *count</code> | Displays the number of elements in the MediaItem. |
| <code>item</code> | <code>{MediaItem} *item(variant Index)</code> | Returns the object with the specified index or name. Type: Long Integer or String. |

MenuActionEvents

Object available in ID 7

| MENUACTIONEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|---|
| <code>afterInvoke</code> | <code>const afterInvoke(afterInvoke)</code> | Dispatched after the MenuAction is invoked. This event does not bubble. This event is not cancelable. |
| <code>beforeInvoke</code> | <code>const beforeInvoke(beforeInvoke)</code> | Dispatched before the MenuAction is invoked. This event does not bubble. This event is cancelable. |

MenuEvents

Object available in ID 7

| MENEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|--|
| <code>beforeDisplay</code> | <code>const beforeDisplay(beforeDisplay)</code> | Dispatched before the Menu is displayed. This event does not bubble. This event is not cancelable. |

MojikumiTable

Object available in ID 7

| MOJIKUMITABLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|---|--|
| <code>delete</code> | <code>void delete()</code> | Deletes the MojikumiTable. |
| <code>extractLabel</code> | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| <code>insertLabel</code> | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |

| MOJIKUMITABLE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|--|--|
| <code>basedOnMojikumiSet</code> | <code>#MojikumiTableDefaults basedOnMojikumiSet</code> | <i>The existing mojikumi set on which to base the new mojikumi set.</i> |
| <code>id</code> | <code>integer •id</code> | <i>The unique ID of the MojikumiTable.</i> |
| <code>index</code> | <code>integer •index</code> | <i>The index of the MojikumiTable within its containing object.</i> |
| <code>label</code> | <code>string label</code> | <i>A property that can be set to any string.</i> |
| <code>name</code> | <code>string name</code> | <i>The name of the MojikumiTable.</i> |
| <code>overrideMojikumiAkiList</code> | <code>variant overrideMojikumiAkiList</code> | <i>The mojikumi overrides for aki. Type: Array of Ordered array containing TargetMojikumiClass:Integer, SideMojikumiClass:Integer, SidesAfterTarget:Boolean, Minimum:Double, Desired:Double, Maximum:Double, CompressionPriority:Integer, AkiDoesNotFloat:Boolean.</i> |
| <code>parent</code> | <code>variant •parent</code> | <i>The parent of the MojikumiTable (a Document or Application). Type: Document or Application.</i> |

MojikumiTables

Object available in ID 7

| MOJIKUMITABLES CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|---|
| <code>add</code> | <code>{MojikumiTable} add(string Name)</code> | <i>Creates a new mojikumi table. / Return value: The new mojikumi table. / Name: The name of the new mojikumi table.</i> |
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | <i>Returns any object in the collection. / Return value: The object.</i> |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | <i>Returns the first object in the collection. / Return value: The first object.</i> |
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | <i>Returns the object with the specified ID. / Return value: The object. / ID: The ID.</i> |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | <i>Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String.</i> |
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | <i>Returns the last object in the collection. / Return value: The last object.</i> |
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | <i>Returns the middle object in the collection. / Return value: The middle object.</i> |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | <i>Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object.</i> |
| <code>previousItem</code> | <code>IDispatch previousItem(IDispatch Obj)</code> | <i>Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object.</i> |
| <code>count</code> | <code>integer •count</code> | <i>Displays the number of elements in the MojikumiTable.</i> |
| <code>item</code> | <code>{MojikumiTable} •item(variant Index)</code> | <i>Returns the object with the specified index or name. Type: Long Integer or String.</i> |

MojikumiUiPreference

Object available in ID 7

| MOJIKUMIUIPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|--|
| <code>lineEndAllOneEm</code> | <code>boolean lineEndAllOneEm</code> | If true, uses full-width spacing for all characters. |
| <code>lineEndAllOneHalfEm</code> | <code>boolean lineEndAllOneHalfEm</code> | If true, uses half-width spacing for all characters. |
| <code>lineEndPeriodOneEm</code> | <code>boolean lineEndPeriodOneEm</code> | If true, uses full-width spacing for punctuation and for the last character in the line. |
| <code>lineEndUkeNoFloat</code> | <code>boolean lineEndUkeNoFloat</code> | If true, uses line end uke no float. |
| <code>oneEmIndentLineEndAllNoFloat</code> | <code>boolean oneEmIndentLineEndAllNoFloat</code> | If true, indents lines one full space and uses no float for all characters. |
| <code>oneEmIndentLineEndAllOneEm</code> | <code>boolean oneEmIndentLineEndAllOneEm</code> | If true, indents lines one full space and uses full-width spacing for all characters. |
| <code>oneEmIndentLineEndAllOneHalfEm</code> | <code>boolean oneEmIndentLineEndAllOneHalfEm</code> | If true, indents lines one full space and uses half-width spacing for all characters. |
| <code>oneEmIndentLineEndPeriodOneEm</code> | <code>boolean oneEmIndentLineEndPeriodOneEm</code> | If true, indents lines one space and uses full-width spacing for punctuation and for the last character in the line. |
| <code>oneEmIndentLineEndUkeNoFloat</code> | <code>boolean oneEmIndentLineEndUkeNoFloat</code> | If true, indents lines one full space and uses line end uke no float. |
| <code>oneEmIndentLineEndUkeOneHalfEm</code> | <code>boolean oneEmIndentLineEndUkeOneHalfEm</code> | If true, indents lines one space and uses line end uke one half space. |
| <code>oneOrOneHalfEmIndentLineEndAllOneEm</code> | <code>boolean oneOrOneHalfEmIndentLineEndAllOneEm</code> | If true, Uses full-wtih spacing for all characters except the last character in the line, which uses either full- or half-width spacing. |
| <code>oneOrOneHalfEmIndentLineEndPeriodOneEm</code> | <code>boolean oneOrOneHalfEmIndentLineEndPeriodOneEm</code> | If true, indents lines one or one-half space and uses full-width spacing for punctuation and for the last character in the line. |
| <code>oneOrOneHalfEmIndentLineEndUkeNoFloat</code> | <code>boolean oneOrOneHalfEmIndentLineEndUkeNoFloat</code> | If true, indents lines one half space or one full space and uses line end uke no float. |
| <code>oneOrOneHalfEmIndentLineEndUkeOneHalfEm</code> | <code>boolean oneOrOneHalfEmIndentLineEndUkeOneHalfEm</code> | If true, indents lines one full or half space and uses line end uke one half space. |
| <code>parent</code> | <code>variant *parent</code> | The parent of the MojikumiUiPreference (a Document or Application). Type: Document or Application. |

MotionPreset

Object available in ID 7

| MOTIONPRESET CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|---|--|
| <code>delete</code> | <code>void delete()</code> | Deletes the MotionPreset. |
| <code>duplicate</code> | <code>{MotionPreset} duplicate(variant Name)</code> | Duplicates the motion preset. / Return value: The duplicated motion preset. / Name: The name for the duplicated motion preset. Type: String. |
| <code>extractLabel</code> | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| <code>insertLabel</code> | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| <code>saveACopy</code> | <code>void saveACopy(string To)</code> | Save a copy of this motion preset to a InDesign motion preset file. / To: The Flash motion preset file to export to. |

| MOTIONPRESET CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|------------------------------|---|
| contents | <code>string contents</code> | <i>Motion preset raw data.</i> |
| id | <code>integer •id</code> | <i>The unique ID of the MotionPreset.</i> |
| index | <code>integer •index</code> | <i>The index of the MotionPreset within its containing object.</i> |
| label | <code>string label</code> | <i>A property that can be set to any string.</i> |
| name | <code>string name</code> | <i>The name of the MotionPreset.</i> |
| parent | <code>variant •parent</code> | <i>The parent of the MotionPreset (a Application or Document). Type: Application or Document.</i> |

MotionPresets

Object available in ID 7

| MOTIONPRESETS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|---|
| add | <code>{MotionPreset} add()</code> | <i>Creates a new MotionPreset. / Return value: The new MotionPreset.</i> |
| anyItem | <code>IDispatch anyItem()</code> | <i>Returns any object in the collection. / Return value: The object.</i> |
| firstItem | <code>IDispatch firstItem()</code> | <i>Returns the first object in the collection. / Return value: The first object.</i> |
| itemByID | <code>IDispatch itemByID(integer Id)</code> | <i>Returns the object with the specified ID. / Return value: The object. / ID: The ID.</i> |
| itemByRange | <code>{Objects} itemByRange(variant From, variant To)</code> | <i>Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String.</i> |
| lastItem | <code>IDispatch lastItem()</code> | <i>Returns the last object in the collection. / Return value: The last object.</i> |
| middleItem | <code>IDispatch middleItem()</code> | <i>Returns the middle object in the collection. / Return value: The middle object.</i> |
| nextItem | <code>IDispatch nextItem(IDispatch Obj)</code> | <i>Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object.</i> |
| previousItem | <code>IDispatch previousItem(IDispatch Obj)</code> | <i>Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object.</i> |
| count | <code>integer •count</code> | <i>Displays the number of elements in the MotionPreset.</i> |
| item | <code>{MotionPreset} •item(variant Index)</code> | <i>Returns the object with the specified index or name. Type: Long Integer or String.</i> |

MovieEvents

Object available in ID 7

| MOVIEEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------|--|--|
| afterAttributeChanged | <code>const afterAttributeChanged(afterAttributeChanged)</code> | Dispatched when the value of a property changes on this Movie. This event bubbles. This event is not cancelable. |
| afterPlace | <code>const afterPlace(afterPlace)</code> | Dispatched after a Movie is placed. This event bubbles. This event is not cancelable. |
| beforePlace | <code>const beforePlace(beforePlace)</code> | Dispatched before a Movie is placed. This event bubbles. This event is cancelable. |

MultiStateObject

Object available in ID 7

| MULTISTATEOBJECT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|--|
| addItemsAsState | <code>void addItemsAsState(Objects PageItems)</code> | Adds items to a specific appearance state of this object. / Pageitems: One or more page items to add as a state. |
| applyObjectStyle | <code>void applyObjectStyle(ObjectStyle Using, boolean ClearingOverrides, boolean ClearingOverridesThroughRootObjectStyle)</code> | Applies the specified object style. / Using: The object style to apply. / ClearingOverrides: If true, clears the MultiStateObject's existing attributes before applying the style. / ClearingOverridesThroughRootObjectStyle: If true, clears attributes and formatting applied to the MultiStateObject that are not defined in the object style. |
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| autoTag | <code>void autoTag()</code> | Tag the object or the parent story using default tags defined in XML preference. |
| bringForward | <code>void bringForward()</code> | Brings the MultiStateObject forward one level in its layer. |
| bringToFront | <code>void bringToFront(variant Reference)</code> | Brings the MultiStateObject to the front of its layer or in front of a particular item. / Reference: The reference object to bring the object in front of (must have same parent). Type: PageItem. |
| changeObject | <code>{Objects} changeObject(variant ReverseOrder)</code> | Finds objects that match the find what value and replace the objects with the change to value. / Return value: The replaced object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| clearObjectStyleOverrides | <code>void clearObjectStyleOverrides()</code> | Clear overrides for object style. |
| clearTransformations | <code>void clearTransformations()</code> | Clears transformations from the MultiStateObject. Transformations include rotation, scaling, flipping, fitting, and shearing. |

| MULTISTATEOBJECT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|--|
| convertShape | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | Converts the MultiStateObject to a different shape. / Given: The MultiStateObject's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0) . Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points). |
| delete | <code>void delete()</code> | Deletes the MultiStateObject. |
| detach | <code>void detach()</code> | Detaches an overridden master page item from the master page. |
| duplicate | <code>{PageItem} duplicate(variant To, variant By)</code> | Duplicates the MultiStateObject at the specified location or offset. / Return value: The duplicated MultiStateObject. / To: The location of the new MultiStateObject, specified in coordinates in the format [x, y]. Type: Array of 2 Units (Doubles or Strings), Spread, Page or Layer. / By: Amount by which to offset the new MultiStateObject from the original MultiStateObject's position. Type: Array of 2 Units (Doubles or Strings). |
| export | <code>void export(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | Exports the object(s) to a file. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| extractLabel | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| findObject | <code>{Objects} findObject(variant ReverseOrder)</code> | Finds objects that match the find what value. / Return value: The found object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| fit | <code>void fit(#FitOptions Given)</code> | Applies the specified fit option to content in a frame. / Given: The fit option to use. |
| flipItem | <code>void flipItem(#Flip Given, variant Around)</code> | Flips the MultiStateObject. / Given: The axis around which to flip the MultiStateObject. / Around: The point around which to flip the MultiStateObject. Type: Array of 2 Units (Doubles or Strings) or idAnchorPoint enumerator. |
| insertLabel | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| markup | <code>void markup(XMLElement Using)</code> | Associates the page item with the specified XML element while preserving existing content. / Using: The XML element. |
| move | <code>void move(variant To, variant By)</code> | Moves the MultiStateObject to a new location. Note: Either the 'to' or 'by' parameter is required; if both parameters are defined, only the to value is used. / To: The new location of the MultiStateObject, in the format (x, y). Type: Array of 2 Units (Doubles or Strings), Spread, Page or Layer. / By: The amount (in measurement units) to move the MultiStateObject relative to its current position, in the format (x, y). Type: Array of 2 Units (Doubles or Strings). |
| override | <code>variant override(Page DestinationPage)</code> | Overrides a master page item and places the item on the document page as a new object. / Return value: The new object. Type: PageItem, Guide, Graphic, Movie or Sound. / DestinationPage: The document page that contains the master page item to override. |
| placeXML | <code>void placeXML(XMLElement Using)</code> | Places XML content into the specified object. Note: Replaces any existing content. / Using: The XML element whose content you want to place. |

| MULTISTATEOBJECT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|--|
| redefineScaling | <code>void redefineScaling(variant To)</code> | Apply an item's scaling to its content if possible. / To: The scale factors to be left on the item. The default is {1.0, 1.0}. Type: Array of 2 Doubles. |
| reframe | <code>void reframe(variant In, variant OpposingCorners)</code> | Move the bounding box of the page item. / In: The bounding box to resize. Type: idCoordinateSpaces enumerator or Ordered array containing CoordinateSpace:idCoordinateSpaces enumerator, BoundsKind:idBoundingBoxLimits enumerator. / OpposingCorners: Opposing corners of new bounding box in the given coordinate space. Type: Array of 2 Arrays of 2 Doubles. |
| releaseAsObjects | <code>void releaseAsObjects()</code> | Releases all the states associated with this object and then destroys the parent object. |
| removeOverride | <code>void removeOverride()</code> | Removes the override from a previously overridden master page item. |
| resize | <code>void resize(variant In, variant From, #ResizeMethods By, variant Values, boolean ResizeIndividually, boolean ConsideringRulerUnits)</code> | Resize the page item. / In: The bounding box to resize. Type: idCoordinateSpaces enumerator, idBoundingBoxLimits enumerator or Ordered array containing CoordinateSpace:idCoordinateSpaces enumerator, BoundsKind:idBoundingBoxLimits enumerator. / From: The transform origin. Legal specifications: relative to bounding box: anchor {anchor {x,y}, bounds kind [, coordinate space]}; relative to coordinate space: {x,y} {x,y} [, coordinate space]; relative to layout window ruler: {x,y}, page index bounds kind. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / By: How the current dimensions are affected by the given values. / Values: The width and height values. Legal dimensions specifications: {x, y [, coordinate space]}, {x, resize constraint [, coordinate space]}, or {resize constraint, y [, coordinate space]}; where x and y are real numbers and coordinate space is used to determine _only_ the unit of length for x and y; coordinate space is ignored for the 'current dimensions times' resize method. Type: Array of Doubles, idResizeConstraints enumerators or idCoordinateSpaces enumerators. / ResizeIndividually: If false and multiple page items are targeted, the new dimensions are attained only by moving the individual items rather than resizing them. / ConsideringRulerUnits: If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| resolve | <code>variant resolve(variant Location, #CoordinateSpaces In, boolean ConsideringRulerUnits)</code> | Get the coordinates of the given location in the specified coordinate system. / Return value: The resolved location for each of the targeted page items. Type: Array of 2 Doubles or Array of Arrays of 2 Doubles. / Location: The location requested. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / In: The coordinate space to use. / ConsideringRulerUnits: If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| select | <code>void select(#SelectionOptions ExistingSelection)</code> | Selects the object. / ExistingSelection: The selection status of the MultiStateObject in relation to previously selected objects. |
| sendBackward | <code>void sendBackward()</code> | Sends the MultiStateObject back one level in its layer. |
| sendToBack | <code>void sendToBack(variant Reference)</code> | Sends the MultiStateObject to the back of its layer or behind a particular item (must have same parent). / Reference: The reference object to send the object behind. Type: PageItem. |
| store | <code>{Asset} store(Library Using)</code> | Stores the object in the specified library. / Return value: The new library asset(s). / Using: The library in which to store the object. |

| MULTISTATEOBJECT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| transform | <code>void transform(#CoordinateSpaces In, variant From, variant WithMatrix, variant ReplacingCurrent, boolean ConsideringRulerUnits)</code> | Transform the page item. / In: The coordinate space to use. / From: The temporary origin during the transformation. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / WithMatrix: Transform matrix. Type: Array of 6 Doubles or TransformationMatrix. / ReplacingCurrent: Transform components to consider; providing this optional parameter causes the target's existing transform components to be replaced with new values. Without this parameter, the given matrix is concatenated onto the target's existing transform combining the effect of the two. Type: idMatrixContent enumerator, Array of idMatrixContent enumerators or Long Integer. / ConsideringRulerUnits: If true then a ruler based origin is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| transformAgain | <code>variant transformAgain()</code> | Transforms the MultiStateObject using the last transformation performed on any object. Transformations include moving, rotating, shearing, scaling, and flipping. . / Return value: Playback errors. Type: Array of Strings. |
| transformAgainIndividually | <code>variant transformAgainIndividually()</code> | Transforms the MultiStateObject using the last transformation performed on any MultiStateObject. Transformations include moving, rotating, shearing, scaling, and flipping. . / Return value: Playback errors. Type: Array of Strings. |
| transformSequenceAgain | <code>variant transformSequenceAgain()</code> | Transforms the MultiStateObject using the last sequence of transform operations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformSequenceAgainIndividually | <code>variant transformSequenceAgainIndividually()</code> | Transforms the MultiStateObject using the last sequence of transformations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings. |
| transformValuesOf | <code>{Objects} transformValuesOf(#CoordinateSpaces In)</code> | Get the transformation values of the page item. / Return value: A list of the transformation matrices. / In: The coordinate space to use. |
| absoluteFlip | <code>#Flip absoluteFlip</code> | Indicates whether the MultiStateObject has been flipped independently of its parent object and, if yes, the direction in which the MultiStateObject was flipped. |
| absoluteHorizontalScale | <code>double absoluteHorizontalScale</code> | The horizontal scale of the MultiStateObject relative to its containing object. |
| absoluteRotationAngle | <code>double absoluteRotationAngle</code> | The rotation angle of the MultiStateObject relative to its containing object. (Range: -360 to 360). |
| absoluteShearAngle | <code>double absoluteShearAngle</code> | The skewing angle of the MultiStateObject relative to its containing object. (Range: -360 to 360). |
| absoluteVerticalScale | <code>double absoluteVerticalScale</code> | The vertical scale of the MultiStateObject relative to its containing object. |
| activeStateIndex | <code>integer activeStateIndex</code> | The index of the active state in the object's states collection. |
| allGraphics | <code>{Objects} *allGraphics</code> | Lists all graphics contained by the MultiStateObject. |
| allowOverrides | <code>boolean allowOverrides</code> | If true, the master page item can be overridden. |
| allPageItems | <code>{Objects} *allPageItems</code> | Lists all page items contained by the MultiStateObject. |
| anchoredObjectSettings | <code>{AnchoredObjectSetting} *anchoredObjectSettings</code> | Anchored object settings. |
| animationSettings | <code>{AnimationSetting} *animationSettings</code> | The page item animation settings. |

| MULTISTATEOBJECT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|--|---|
| <code>appliedObjectStyle</code> | <code>{ObjectStyle} appliedObjectStyle</code> | <i>The object style applied to the MultiStateObject.</i> |
| <code>associatedXMLElement</code> | <code>{XMLElement} *associatedXMLElement</code> | <i>The XML element associated with the MultiStateObject.</i> |
| <code>bottomLeftCornerOption</code> | <code>#CornerOptions bottomLeftCornerOption</code> | <i>The shape to apply to the bottom left corner of rectangular shapes.</i> |
| <code>bottomLeftCornerRadius</code> | <code>variant bottomLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>bottomRightCornerOption</code> | <code>#CornerOptions bottomRightCornerOption</code> | <i>The shape to apply to the bottom right corner of rectangular shapes.</i> |
| <code>bottomRightCornerRadius</code> | <code>variant bottomRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>contentTransparencySettings</code> | <code>{ContentTransparencySetting} *contentTransparencySettings</code> | <i>Transparency settings for the content of the MultiStateObject.</i> |
| <code>description</code> | <code>string description</code> | <i>The description of the MultiStateObject.</i> |
| <code>endCap</code> | <code>#EndCap endCap</code> | <i>The end shape of an open path.</i> |
| <code>endJoin</code> | <code>#EndJoin endJoin</code> | <i>The corner join applied to the MultiStateObject.</i> |
| <code>epss</code> | <code>{EPSs} *epss</code> | <i>A collection of EPS files.</i> |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |
| <code>fillColor</code> | <code>variant fillColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the fill of the MultiStateObject. . Type: Swatch or String.</i> |
| <code>fillTint</code> | <code>double fillTint</code> | <i>The percent of tint to use in the MultiStateObject's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>fillTransparencySettings</code> | <code>{FillTransparencySetting} *fillTransparencySettings</code> | <i>Transparency settings for the fill applied to the MultiStateObject.</i> |
| <code>flip</code> | <code>#Flip flip</code> | <i>The direction in which to flip the printed image.</i> |
| <code>gapColor</code> | <code>{Swatch} gapColor</code> | <i>The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type.</i> |
| <code>gapTint</code> | <code>double gapTint</code> | <i>The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.).</i> |
| <code>geometricBounds</code> | <code>variant geometricBounds</code> | <i>The bounds of the MultiStateObject excluding the stroke width, in the format [y1, x1, y2, x2], which give the coordinates of the top-left and bottom-right corners of the bounding box. Type: Array of 4 Units (Doubles or Strings).</i> |
| <code>gradientFillAngle</code> | <code>double gradientFillAngle</code> | <i>The angle of a linear gradient applied to the fill of the MultiStateObject. (Range: -180 to 180).</i> |
| <code>gradientFillLength</code> | <code>variant gradientFillLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the MultiStateObject. Type: Unit (Double or String).</i> |
| <code>gradientFillStart</code> | <code>variant gradientFillStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the fill of the MultiStateObject, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| <code>gradientStrokeAngle</code> | <code>double gradientStrokeAngle</code> | <i>The angle of a linear gradient applied to the stroke of the MultiStateObject. (Range: -180 to 180).</i> |
| <code>gradientStrokeLength</code> | <code>variant gradientStrokeLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the MultiStateObject. Type: Unit (Double or String).</i> |

| MULTISTATEOBJECT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|--|--|
| gradientStrokeStart | <i>variant</i> gradientStrokeStart | The starting point (in page coordinates) of a gradient applied to the stroke of the MultiStateObject, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings). |
| graphicLines | {GraphicLines} •graphicLines | A collection of graphic lines. |
| graphics | {Graphics} •graphics | A collection of imported graphics in any graphic file format (vector, metafile, or bitmap). |
| groups | {Groups} •groups | A collection of groups. |
| horizontalScale | double horizontalScale | The horizontal scaling applied to the MultiStateObject. |
| id | integer •id | The unique ID of the MultiStateObject. |
| images | {Images} •images | A collection of bitmap images in any bitmap file format (including TIFF, JPEG, or GIF). |
| index | integer •index | The index of the MultiStateObject within its containing object. |
| initiallyHidden | boolean initiallyHidden | Determines if this object is initially hidden when displayed in an exported SWF file. |
| itemLayer | {Layer} itemLayer | The layer that the MultiStateObject is on. |
| label | string label | A property that can be set to any string. |
| leftLineEnd | #ArrowHead leftLineEnd | The arrowhead applied to the start of the path. |
| localDisplaySetting | #DisplaySettingOptions localDisplaySetting | Display performance options for the MultiStateObject. |
| locked | boolean locked | If true, the MultiStateObject is locked. |
| miterLimit | double miterLimit | The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500). |
| name | string name | The name of the MultiStateObject. |
| nonprinting | boolean nonprinting | If true, the MultiStateObject does not print. |
| ovals | {Ovals} •ovals | A collection of ellipses. |
| overprintFill | boolean overprintFill | If true, the MultiStateObject's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors. |
| overprintGap | boolean overprintGap | If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors. |
| overprintStroke | boolean overprintStroke | If true, the MultiStateObject's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors. |
| overridden | boolean •overridden | If true, the object originated on a master spread and was overridden. If false, the object either originated on a master spread and was not overridden, or the object did not originate on a master page. |
| overriddenMasterPageItem | <i>variant</i> •overriddenMasterPageItem | An object that originated on a master page and has been overridden. Type: PageItem, Guide, Graphic, Movie or Sound. |
| pageItems | {PageItems} •pageItems | The page items collection, which can be used to process all page items in a container (such as a document, page, or group), regardless of type. |
| parent | <i>variant</i> •parent | The parent of the MultiStateObject (a Snippet, PlaceGun, Spread, MasterSpread, SplineItem, Polygon, GraphicLine, Rectangle, Oval, Group or Character). Type: Snippet, PlaceGun, Spread, MasterSpread, SplineItem, Polygon, GraphicLine, Rectangle, Oval, Group or Character. |
| parentPage | {Page} •parentPage | The page on which this page item appears. |

| MULTISTATEOBJECT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|---|---|
| paths | {Paths} *paths | A collection of paths. |
| pdfs | {PDFs} *pdfs | A collection of PDF files. |
| picts | {PICTs} *picts | A collection of PICT graphics. |
| polygons | {Polygons} *polygons | A collection of polygons. |
| preferences | {Preferences} *preferences | A collection of preferences objects. |
| rectangles | {Rectangles} *rectangles | A collection of rectangles. |
| rightLineEnd | #ArrowHead rightLineEnd | The arrowhead applied to the end of the path. |
| rotationAngle | double rotationAngle | The rotation angle of the MultiStateObject. (Range: -360 to 360). |
| shearAngle | double shearAngle | The skewing angle applied to the MultiStateObject. (Range: -360 to 360). |
| splineItems | {SplineItems} *splineItems | The spline items collection. |
| states | {States} *states | A collection of states. |
| strokeAlignment | #StrokeAlignment strokeAlignment | The stroke alignment applied to the MultiStateObject. |
| strokeColor | variant strokeColor | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the MultiStateObject. Type: Swatch or String. |
| strokeCornerAdjustment | #StrokeCornerAdjustment strokeCornerAdjustment | The corner adjustment applied to the MultiStateObject. |
| strokeDashAndGap | variant strokeDashAndGap | The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings). |
| strokeTint | double strokeTint | The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| strokeTransparencySettings | {StrokeTransparencySetting} *strokeTransparencySettings | Transparency settings for the stroke. |
| strokeType | variant strokeType | The name of the stroke style to apply. Type: StrokeStyle or String. |
| strokeWeight | variant strokeWeight | The weight (in points) to apply to the MultiStateObject's stroke. Type: Unit (Double or String). |
| textFrames | {TextFrames} *textFrames | A collection of text frames. |
| textWrapPreferences | {TextWrapPreference} *textWrapPreferences | The text wrap preference properties that define the default formatting for wrapping text around objects. |
| timingSettings | {TimingSetting} *timingSettings | The object timing settings. |
| topLeftCornerOption | #CornerOptions topLeftCornerOption | The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight. |
| topLeftCornerRadius | variant topLeftCornerRadius | The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String). |
| topRightCornerOption | #CornerOptions topRightCornerOption | The shape to apply to the top right corner of rectangular shapes. |
| topRightCornerRadius | variant topRightCornerRadius | The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String). |
| transparencySettings | {TransparencySetting} *transparencySettings | Transparency settings. |

| MULTISTATEOBJECT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|------------------------------------|--|
| <code>verticalScale</code> | <code>double verticalScale</code> | The vertical scaling applied to the MultiStateObject. |
| <code>visible</code> | <code>boolean visible</code> | If true, the MultiStateObject is visible. |
| <code>visibleBounds</code> | <code>variant visibleBounds</code> | The bounds of the MultiStateObject including the stroke width, in the format [y1, x1, y2, x2], which give the coordinates of the top-left and bottom-right corners of the bounding box. Type: Array of 4 Units (Doubles or Strings). |
| <code>wmfs</code> | <code>{WMFs} •wmfs</code> | A collection of WMF graphics. |

MultiStateObjectEvents

Object available in ID 7

| MULTISTATEOBJECTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|--|
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | Dispatched after a MultiStateObject is placed. This event bubbles. This event is not cancelable. |
| <code>beforePlace</code> | <code>const beforePlace(beforePlace)</code> | Dispatched before a MultiStateObject is placed. This event bubbles. This event is cancelable. |

MultiStateObjects

Object available in ID 7

| MULTISTATEOBJECTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|--|---|
| <code>add</code> | <code>{MultiStateObject} add(variant Layer, #LocationOptions At, variant Reference)</code> | Creates a new MultiStateObject. / Return value: The new MultiStateObject. / Layer: The layer on which to create the MultiStateObject. Type: Layer. / At: The location at which to insert the MultiStateObject relative to the reference object or within the container object. / Reference: The reference object. Note: Required when the at parameter is before or after. Type: Document, Spread, MasterSpread, Page, Layer or PageItem. |
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | Returns any object in the collection. / Return value: The object. |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | Returns the first object in the collection. / Return value: The first object. |
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | Returns the object with the specified ID. / Return value: The object. / ID: The ID. |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |

| MULTISTATEOBJECTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|--|---|
| previousItem | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| count | <code>integer *count</code> | Displays the number of elements in the MultiStateObject. |
| item | <code>{MultiStateObject} *item(variant Index)</code> | Returns the object with the specified index or name. Type: Long Integer or String. |

MutationEvent

Object available in ID 7

| MUTATIONEVENT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|---|
| preventDefault | <code>void preventDefault()</code> | Cancels the default behavior of the event on its target. |
| stopPropagation | <code>void stopPropagation()</code> | Stops propagation of the event beyond the current target. |
| attributeName | <code>string *attributeName</code> | The name of the property that changed. |
| attributeValue | <code>variant *attributeValue</code> | The current value of the property that changed. Type: Any Type. |
| bubbles | <code>boolean *bubbles</code> | If true, the event supports the bubbling phase of propagation. |
| cancelable | <code>boolean *cancelable</code> | If true, the default behavior of the event on its target can be canceled. |
| currentTarget | <code>IDispatch *currentTarget</code> | The current propagation target of the event. |
| defaultPrevented | <code>boolean *defaultPrevented</code> | If true, the default behavior of the event on its target has been canceled. |
| eventPhase | <code>#EventPhases *eventPhase</code> | The current propagation phase of the event. |
| eventType | <code>string *eventType</code> | The name of the event. |
| id | <code>integer *id</code> | The unique ID of the MutationEvent. |
| index | <code>integer *index</code> | The index of the MutationEvent within its containing object. |
| parent | <code>variant *parent</code> | The parent of the MutationEvent. (...) |
| propagationStopped | <code>boolean *propagationStopped</code> | If true, propagation of the event beyond the current target has been stopped. |
| target | <code>IDispatch *target</code> | The target of the event. |
| timeStamp | <code>date *timeStamp</code> | The time the event was initialized. |

MutationEventEvents

Object available in ID 7

| MUTATIONEVENTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|--|--|
| afterAttributeChanged | <code>const afterAttributeChanged(afterAttributeChanged)</code> | Dispatched when the value of a property changes on this MutationEvent. This event bubbles. This event is not cancelable. |

NamedGrid

Object available in ID 7

| NAMEDGRID CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|---|--|
| <code>delete</code> | <code>void delete()</code> | Deletes the NamedGrid. |
| <code>extractLabel</code> | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| <code>insertLabel</code> | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| <code>gridData</code> | <code>{GridDataInformation} *gridData</code> | Default grid properties. Note: Applies to named, layout, and frame (story) grids. |
| <code>id</code> | <code>integer *id</code> | The unique ID of the NamedGrid. |
| <code>index</code> | <code>integer *index</code> | The index of the NamedGrid within its containing object. |
| <code>label</code> | <code>string label</code> | A property that can be set to any string. |
| <code>name</code> | <code>string name</code> | The name of the NamedGrid. |
| <code>parent</code> | <code>variant *parent</code> | The parent of the NamedGrid (a Document or Application). Type: Document or Application. |
| <code>preferences</code> | <code>{Preferences} *preferences</code> | A collection of preferences objects. |

NamedGrids

Object available in ID 7

| NAMEDGRIDS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|--|
| <code>add</code> | <code>{NamedGrid} add()</code> | Creates a new NamedGrid. / Return value: The new NamedGrid. |
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | Returns any object in the collection. / Return value: The object. |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | Returns the first object in the collection. / Return value: The first object. |
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | Returns the object with the specified ID. / Return value: The object. / ID: The ID. |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| <code>previousItem</code> | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| <code>count</code> | <code>integer *count</code> | Displays the number of elements in the NamedGrid. |

| NAMEDGRIDS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------|----------------------------------|--|
| item | {NamedGrid} •item(variant Index) | Returns the object with the specified index or name. Type: Long Integer or String. |

NavigationPoint

Object available in ID 7

| NAVIGATIONPOINT CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|-----------------|---|
| delete | void delete() | Deletes the NavigationPoint. |
| id | integer •id | Unique internally-generated identifier (read only). |
| index | integer •index | The index of the NavigationPoint within its containing object. |
| name | string name | The UI display name. |
| parent | {Movie} •parent | The parent of the NavigationPoint (a Movie). |
| time | double time | The time in seconds rounded up to two decimal places (e.g., 3.115 rounded to 3.12). |

NavigationPoints

Object available in ID 7

| NAVIGATIONPOINTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|---|--|
| add | {NavigationPoint} add() | Creates a new NavigationPoint. / Return value: The new NavigationPoint. |
| anyItem | IDispatch anyItem() | Returns any object in the collection. / Return value: The object. |
| firstItem | IDispatch firstItem() | Returns the first object in the collection. / Return value: The first object. |
| itemByRange | {Objects} itemByRange(variant From, variant To) | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| lastItem | IDispatch lastItem() | Returns the last object in the collection. / Return value: The last object. |
| middleItem | IDispatch middleItem() | Returns the middle object in the collection. / Return value: The middle object. |
| nextItem | IDispatch nextItem(IDispatch Obj) | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| previousItem | IDispatch previousItem(IDispatch Obj) | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| count | integer •count | Displays the number of elements in the NavigationPoint. |
| item | {NavigationPoint} •item(variant Index) | Returns the object with the specified index or name. Type: Long Integer or String. |

OvalEvents

Object available in ID 7

| OVALEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------|--|---|
| afterPlace | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a Oval is placed. This event bubbles. This event is not cancelable.</i> |
| beforePlace | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a Oval is placed. This event bubbles. This event is cancelable.</i> |

PageItemEvents

Object available in ID 7

| PAGEITEMEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|---|
| afterPlace | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a Pageitem is placed. This event bubbles. This event is not cancelable.</i> |
| beforePlace | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a Pageitem is placed. This event bubbles. This event is cancelable.</i> |

PDFEvents

Object available in ID 7

| PDFEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------|--|--|
| afterPlace | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a PDF is placed. This event bubbles. This event is not cancelable.</i> |
| beforePlace | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a PDF is placed. This event bubbles. This event is cancelable.</i> |

PICTEvents

Object available in ID 7

| PICTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------|--|---|
| afterPlace | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a PICT is placed. This event bubbles. This event is not cancelable.</i> |
| beforePlace | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a PICT is placed. This event bubbles. This event is cancelable.</i> |

PlaceGun

Object available in ID 7

| PLACEGUN CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------|--|---|
| abortPlaceGun | <code>void abortPlaceGun()</code> | Delete the contents of the place gun. |
| loadPlaceGun | <code>void loadPlaceGun(variant fileName, boolean ShowingOptions)</code> | Load the place gun with one or more files. / <i>FileName</i> : One or more files to place. Type: File (String) or Array of Files (Strings). / <i>ShowingOptions</i> : Whether to display the import options dialog. |
| rotate | <code>void rotate(#RotationDirection Direction)</code> | Rotate the contents of the place gun. / <i>Direction</i> : Which direction to rotate the contents. |
| buttons | <code>{Buttons} *buttons</code> | A collection of buttons. |
| epss | <code>{EPSS} *epss</code> | A collection of EPS files. |
| epsTexts | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| formFields | <code>{FormFields} *formFields</code> | A collection of form fields. |
| graphicLines | <code>{GraphicLines} *graphicLines</code> | A collection of graphic lines. |
| graphics | <code>{Graphics} *graphics</code> | A collection of imported graphics in any graphic file format (vector, metafile, or bitmap). |
| groups | <code>{Groups} *groups</code> | A collection of groups. |
| images | <code>{Images} *images</code> | A collection of bitmap images in any bitmap file format (including TIFF, JPEG, or GIF). |
| importedPages | <code>{ImportedPages} *importedPages</code> | Imported InDesign pages. |
| loaded | <code>boolean *loaded</code> | Whether the place gun is currently loaded with content for placing. |
| multiStateObjects | <code>{MultiStateObjects} *multiStateObjects</code> | A collection of multi-state objects. |
| ovals | <code>{Ovals} *ovals</code> | A collection of ellipses. |
| pageItems | <code>{PageItems} *pageItems</code> | The page items collection, which can be used to process all page items in a container (such as a document, page, or group), regardless of type. |
| parent | <code>{Document} *parent</code> | The parent of the PlaceGun (a Document). |
| pdfs | <code>{PDFs} *pdfs</code> | A collection of PDF files. |
| picts | <code>{PICTs} *picts</code> | A collection of PICT graphics. |
| polygons | <code>{Polygons} *polygons</code> | A collection of polygons. |
| rectangles | <code>{Rectangles} *rectangles</code> | A collection of rectangles. |
| snippets | <code>{Snippets} *snippets</code> | A collection of IDML snippets. |
| splineItems | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| textFrames | <code>{TextFrames} *textFrames</code> | A collection of text frames. |
| wmfs | <code>{WMFs} *wmfs</code> | A collection of WMF graphics. |

PolygonEvents

Object available in ID 7

| POLYGONEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|---|
| afterPlace | <code>const afterPlace(afterPlace)</code> | Dispatched after a Polygon is placed. This event bubbles. This event is not cancelable. |
| beforePlace | <code>const beforePlace(beforePlace)</code> | Dispatched before a Polygon is placed. This event bubbles. This event is cancelable. |

PrintEventEvents

Object available in ID 7

| PRINTEVENTEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|---|
| afterPrint | <code>const afterPrint(afterPrint)</code> | Dispatched after a PrintEvent is printed. This event bubbles. This event is not cancelable. |
| beforePrint | <code>const beforePrint(beforePrint)</code> | Dispatched before a PrintEvent is printed. This event bubbles. This event is cancelable. |

RectangleEvents

Object available in ID 7

| RECTANGLEEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|--|---|
| afterPlace | <code>const afterPlace(afterPlace)</code> | Dispatched after a Rectangle is placed. This event bubbles. This event is not cancelable. |
| beforePlace | <code>const beforePlace(beforePlace)</code> | Dispatched before a Rectangle is placed. This event bubbles. This event is cancelable. |

ScriptMenuActionEvents

Object available in ID 7

| SCRIPTMENUACTIONEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|--|
| afterInvoke | <code>const afterInvoke(afterInvoke)</code> | Dispatched after the ScriptMenuAction is invoked. This event does not bubble. This event is not cancelable. |
| beforeDisplay | <code>const beforeDisplay(beforeDisplay)</code> | Dispatched before the ScriptMenuAction is displayed. This event does not bubble. This event is not cancelable. |
| beforeInvoke | <code>const beforeInvoke(beforeInvoke)</code> | Dispatched before the ScriptMenuAction is invoked. This event does not bubble. This event is cancelable. |
| onInvoke | <code>const onInvoke(onInvoke)</code> | Dispatched when the ScriptMenuAction is invoked. This event does not bubble. This event is not cancelable. |

Snippet

Object available in ID 7

| SNIPPET CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------|---|--|
| <code>delete</code> | <code>void delete()</code> | <i>Deletes the Snippet.</i> |
| <code>extractLabel</code> | <code>string extractLabel(string Key)</code> | <i>Gets the label value associated with the specified key. / Return value: The label. / Key: The key.</i> |
| <code>insertLabel</code> | <code>void insertLabel(string Key, string Value)</code> | <i>Sets the label to the value associated with the specified key. / Key: The key. / Value: The value.</i> |
| <code>buttons</code> | <code>{Buttons} *buttons</code> | <i>A collection of buttons.</i> |
| <code>epss</code> | <code>{EPSs} *epss</code> | <i>A collection of EPS files.</i> |
| <code>epsTexts</code> | <code>{EPSTexts} *epsTexts</code> | <i>EPSTexts.</i> |
| <code>formFields</code> | <code>{FormFields} *formFields</code> | <i>A collection of form fields.</i> |
| <code>graphicLines</code> | <code>{GraphicLines} *graphicLines</code> | <i>A collection of graphic lines.</i> |
| <code>graphics</code> | <code>{Graphics} *graphics</code> | <i>A collection of imported graphics in any graphic file format (vector, metafile, or bitmap).</i> |
| <code>groups</code> | <code>{Groups} *groups</code> | <i>A collection of groups.</i> |
| <code>id</code> | <code>integer *id</code> | <i>The unique ID of the Snippet.</i> |
| <code>images</code> | <code>{Images} *images</code> | <i>A collection of bitmap images in any bitmap file format (including TIFF, JPEG, or GIF).</i> |
| <code>index</code> | <code>integer *index</code> | <i>The index of the Snippet within its containing object.</i> |
| <code>label</code> | <code>string label</code> | <i>A property that can be set to any string.</i> |
| <code>multiStateObjects</code> | <code>{MultiStateObjects} *multiStateObjects</code> | <i>A collection of multi-state objects.</i> |
| <code>name</code> | <code>string name</code> | <i>The name of the Snippet; this is an alias to the Snippet's label property.</i> |
| <code>ovals</code> | <code>{Ovals} *ovals</code> | <i>A collection of ellipses.</i> |
| <code>pageItems</code> | <code>{PageItems} *pageItems</code> | <i>The page items collection, which can be used to process all page items in a container (such as a document, page, or group), regardless of type.</i> |
| <code>parent</code> | <code>{PlaceGun} *parent</code> | <i>The parent of the Snippet (a PlaceGun).</i> |
| <code>pdfs</code> | <code>{PDFs} *pdfs</code> | <i>A collection of PDF files.</i> |
| <code>picts</code> | <code>{PICTs} *picts</code> | <i>A collection of PICT graphics.</i> |
| <code>polygons</code> | <code>{Polygons} *polygons</code> | <i>A collection of polygons.</i> |
| <code>rectangles</code> | <code>{Rectangles} *rectangles</code> | <i>A collection of rectangles.</i> |
| <code>splineItems</code> | <code>{SplineItems} *splineItems</code> | <i>The spline items collection.</i> |
| <code>textFrames</code> | <code>{TextFrames} *textFrames</code> | <i>A collection of text frames.</i> |
| <code>wmfs</code> | <code>{WMFs} *wmfs</code> | <i>A collection of WMF graphics.</i> |

SnippetEvents

Object available in ID 7

| SNIPPETEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|---|
| <code>beforePlace</code> | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a Snippet is placed. This event bubbles. This event is cancelable.</i> |

Snippets

Object available in ID 7

| SNIPPETS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|---|
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | <i>Returns any object in the collection. / Return value: The object.</i> |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | <i>Returns the first object in the collection. / Return value: The first object.</i> |
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | <i>Returns the object with the specified ID. / Return value: The object. / ID: The ID.</i> |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | <i>Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String.</i> |
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | <i>Returns the last object in the collection. / Return value: The last object.</i> |
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | <i>Returns the middle object in the collection. / Return value: The middle object.</i> |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | <i>Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object.</i> |
| <code>previousItem</code> | <code>IDispatch previousItem(IDispatch Obj)</code> | <i>Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object.</i> |
| <code>count</code> | <code>integer *count</code> | <i>Displays the number of elements in the Snippet.</i> |
| <code>item</code> | <code>{Snippet} *item(variant Index)</code> | <i>Returns the object with the specified index or name. Type: Long Integer or String.</i> |

SoundEvents

Object available in ID 7

| SOUNDEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|---|
| <code>afterAttributeChanged</code> | <code>const afterAttributeChanged(afterAttributeChanged)</code> | <i>Dispatched when the value of a property changes on this Sound. This event bubbles. This event is not cancelable.</i> |
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a Sound is placed. This event bubbles. This event is not cancelable.</i> |
| <code>beforePlace</code> | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a Sound is placed. This event bubbles. This event is cancelable.</i> |

SplineItem

Object available in ID 7

| SPLINEITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------------|--|---|
| addPath | <code>{PageItem} addPath(Objects With)</code> | <i>Creates a new page item by combining the SplineItem with other objects. / Deletes the objects if they do not intersect. / Return value: The new page item. / With: The object(s) to add.</i> |
| applyObjectStyle | <code>void applyObjectStyle(ObjectStyle Using, boolean ClearingOverrides, boolean ClearingOverridesThroughRootObjectType)</code> | <i>Applies the specified object style. / Using: The object style to apply. / ClearingOverrides: If true, clears the SplineItem's existing attributes before applying the style. / ClearingOverridesThroughRootObjectType: If true, clears attributes and formatting applied to the SplineItem that are not defined in the object style.</i> |
| asynchronousExportFile | <code>{BackgroundTask} asynchronousExportFile(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | <i>asynchronously exports the object(s) to a file. / Return value: object representing the asynchronous export. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| autoTag | <code>void autoTag()</code> | <i>Tag the object or the parent story using default tags defined in XML preference.</i> |
| bringForward | <code>void bringForward()</code> | <i>Brings the SplineItem forward one level in its layer.</i> |
| bringToFront | <code>void bringToFront(variant Reference)</code> | <i>Brings the SplineItem to the front of its layer or in front of a particular item. / Reference: The reference object to bring the object in front of (must have same parent). Type: PageItem.</i> |
| changeObject | <code>{Objects} changeObject(variant ReverseOrder)</code> | <i>Finds objects that match the find what value and replace the objects with the change to value. / Return value: The replaced object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean.</i> |
| checkIn | <code>boolean checkIn(variant VersionComments, boolean ForceSave)</code> | <i>Checks in the story or stories. / Return value: True if the check in succeeded. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version.</i> |
| checkOut | <code>boolean checkOut()</code> | <i>Checks out the story. / Return value: If true, the check out succeeded.</i> |
| clearObjectStyleOverrides | <code>void clearObjectStyleOverrides()</code> | <i>Clear overrides for object style.</i> |
| clearTransformations | <code>void clearTransformations()</code> | <i>Clears transformations from the SplineItem. Transformations include rotation, scaling, flipping, fitting, and shearing.</i> |
| convertShape | <code>void convertShape(#ConvertShapeOptions Given, variant NumberOfSides, variant InsetPercentage, variant CornerRadius)</code> | <i>Converts the SplineItem to a different shape. / Given: The SplineItem's new shape. / NumberOfSides: The number of sides for the resulting polygon. (Range: 3 to 100). Type: Long Integer (3 - 100). / InsetPercentage: The star inset percentage for the resulting polygon. (Range: 0.0 to 100.0) . Type: Long Integer (0 - 100). / CornerRadius: The corner radius of the resulting rectangle. Type: Unit (Double or String) (0 - 800 points).</i> |
| delete | <code>void delete()</code> | <i>Deletes the SplineItem.</i> |
| detach | <code>void detach()</code> | <i>Detaches an overridden master page item from the master page.</i> |

| SPLINEITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|--|
| duplicate | <code>{PageItem} duplicate(variant To, variant By)</code> | Duplicates the SplineItem at the specified location or offset. / Return value: The duplicated SplineItem. / To: The location of the new SplineItem, specified in coordinates in the format [x, y]. Type: Array of 2 Units (Doubles or Strings), Spread, Page or Layer. / By: Amount by which to offset the new SplineItem from the original SplineItem's position. Type: Array of 2 Units (Doubles or Strings). |
| excludeOverlapPath | <code>{PageItem} excludeOverlapPath(Objects With)</code> | Creates a new page item by excluding the overlapping areas of the SplineItem and other objects. / Return value: The new page item. / With: The object(s) to exclude. |
| export | <code>void export(variant Format, string To, boolean ShowingOptions, variant Using, variant VersionComments, boolean ForceSave)</code> | Exports the object(s) to a file. / Format: The export format, specified as an enumeration value or as an extension that appears in the Save as type or Format menu in the Export dialog. Type: idExportFormat enumerator or String. / To: The path to the export file. / ShowingOptions: If true, displays the export options dialog. / Using: The export style. Type: PDFExportPreset. / VersionComments: The comment for this version. Type: String. / ForceSave: If true, forcibly saves a version. |
| extractLabel | <code>string extractLabel(string Key)</code> | Gets the label value associated with the specified key. / Return value: The label. / Key: The key. |
| findObject | <code>{Objects} findObject(variant ReverseOrder)</code> | Finds objects that match the find what value. / Return value: The found object(s). / ReverseOrder: If true, returns the results in reverse order. Type: Boolean. |
| fit | <code>void fit(#FitOptions Given)</code> | Applies the specified fit option to content in a frame. / Given: The fit option to use. |
| flipItem | <code>void flipItem(#Flip Given, variant Around)</code> | Flips the SplineItem. / Given: The axis around which to flip the SplineItem. / Around: The point around which to flip the SplineItem. Type: Array of 2 Units (Doubles or Strings) or idAnchorPoint enumerator. |
| insertLabel | <code>void insertLabel(string Key, string Value)</code> | Sets the label to the value associated with the specified key. / Key: The key. / Value: The value. |
| intersectPath | <code>{PageItem} intersectPath(Objects With)</code> | Creates a new page item by intersecting the SplineItem with other objects. Returns an error if the objects do not intersect. / Return value: The new page item. / With: The object(s) with which to intersect. |
| makeCompoundPath | <code>{PageItem} makeCompoundPath(Objects With)</code> | Creates a compound path by combining the path(s) of the SplineItem with the paths of other objects. / Return value: The new compound path. / With: The other objects whose paths to include in the new compound path. |
| markup | <code>void markup(XMLElement Using)</code> | Associates the page item with the specified XML element while preserving existing content. / Using: The XML element. |
| minusBack | <code>{PageItem} minusBack(Objects With)</code> | Creates a new page item by reverse subtracting the overlapping areas of the SplineItem and other objects. / Return value: The new page item. / With: The object(s) to reverse subtract. |
| move | <code>void move(variant To, variant By)</code> | Moves the SplineItem to a new location. Note: Either the 'to' or 'by' parameter is required; if both parameters are defined, only the to value is used. / To: The new location of the SplineItem, in the format (x, y). Type: Array of 2 Units (Doubles or Strings), Spread, Page or Layer. / By: The amount (in measurement units) to move the SplineItem relative to its current position, in the format (x, y). Type: Array of 2 Units (Doubles or Strings). |
| override | <code>variant override(Page DestinationPage)</code> | Overrides a master page item and places the item on the document page as a new object. / Return value: The new object. Type: PageItem, Guide, Graphic, Movie or Sound. / DestinationPage: The document page that contains the master page item to override. |

| SPLINEITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|---|
| place | <code>variant place(string FileName, boolean ShowingOptions)</code> | Places the file. / Return value: The placed file. Type: Array of PageItems, Graphics, Movies, Sounds, Stories or Texts. / FileName: The file to place. / ShowingOptions: Whether to display the import options dialog. |
| placeXML | <code>void placeXML(XMLElement Using)</code> | Places XML content into the specified object. Note: Replaces any existing content. / Using: The XML element whose content you want to place. |
| redefineScaling | <code>void redefineScaling(variant To)</code> | Apply an item's scaling to its content if possible. / To: The scale factors to be left on the item. The default is {1.0, 1.0}. Type: Array of 2 Doubles. |
| reframe | <code>void reframe(variant In, variant OpposingCorners)</code> | Move the bounding box of the page item. / In: The bounding box to resize. Type: idCoordinateSpaces enumerator or Ordered array containing CoordinateSpace:idCoordinateSpaces enumerator, BoundsKind:idBoundingBoxLimits enumerator. / OpposingCorners: Opposing corners of new bounding box in the given coordinate space. Type: Array of 2 Arrays of 2 Doubles. |
| releaseCompoundPath | <code>{Objects} releaseCompoundPath()</code> | Releases a compound path. / Return value: The paths that made up the compound path (as individual objects). |
| removeOverride | <code>void removeOverride()</code> | Removes the override from a previously overridden master page item. |
| resize | <code>void resize(variant In, variant From, #ResizeMethods By, variant Values, boolean ResizeIndividually, boolean ConsideringRulerUnits)</code> | Resize the page item. / In: The bounding box to resize. Type: idCoordinateSpaces enumerator, idBoundingBoxLimits enumerator or Ordered array containing CoordinateSpace:idCoordinateSpaces enumerator, BoundsKind:idBoundingBoxLimits enumerator. / From: The transform origin. Legal specifications: relative to bounding box: anchor {anchor {x,y}, bounds kind [, coordinate space]}; relative to coordinate space: {x,y} {{x,y} [, coordinate space]}; relative to layout window ruler: {{x,y}, page index bounds kind}. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / By: How the current dimensions are affected by the given values. / Values: The width and height values. Legal dimensions specifications: {x, y [, coordinate space]}, {x, resize constraint [, coordinate space]}, or {resize constraint, y [, coordinate space]}; where x and y are real numbers and coordinate space is used to determine _only_ the unit of length for x and y; coordinate space is ignored for the 'current dimensions times' resize method). Type: Array of Doubles, idResizeConstraints enumerators or idCoordinateSpaces enumerators. / ResizeIndividually: If false and multiple page items are targeted, the new dimensions are attained only by moving the individual items rather than resizing them. / ConsideringRulerUnits: If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| resolve | <code>variant resolve(variant Location, #CoordinateSpaces In, boolean ConsideringRulerUnits)</code> | Get the coordinates of the given location in the specified coordinate system. / Return value: The resolved location for each of the targeted page items. Type: Array of 2 Doubles or Array of Arrays of 2 Doubles. / Location: The location requested. Type: Array of 2 Doubles, idAnchorPoint enumerator or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / In: The coordinate space to use. / ConsideringRulerUnits: If true then a ruler location is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page. |
| revert | <code>boolean revert()</code> | Reverts the document to its state at the last save operation. / Return value: If true, the revert succeeded. |
| select | <code>void select(#SelectionOptions ExistingSelection)</code> | Selects the object. / ExistingSelection: The selection status of the SplineItem in relation to previously selected objects. |

| SPLINEITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| sendBackward | <code>void sendBackward()</code> | <i>Sends the SplineItem back one level in its layer.</i> |
| sendToBack | <code>void sendToBack(variant Reference)</code> | <i>Sends the SplineItem to the back of its layer or behind a particular item (must have same parent). / Reference: The reference object to send the object behind. Type: Pagelitem.</i> |
| store | <code>{Asset} store(Library Using)</code> | <i>Stores the object in the specified library. / Return value: The new library asset(s). / Using: The library in which to store the object.</i> |
| subtractPath | <code>{PageItem} subtractPath(Objects With)</code> | <i>Creates a new page item by subtracting the overlapping areas of the SplineItem and other objects. / Return value: The new page item. / With: The object(s) to subtract.</i> |
| transform | <code>void transform(#CoordinateSpaces In, variant From, variant WithMatrix, variant ReplacingCurrent, boolean ConsideringRulerUnits)</code> | <i>Transform the page item. / In: The coordinate space to use. / From: The temporary origin during the transformation. Type: Array of 2 Doubles, idAnchorPoint enumerators or Array of Arrays of 2 Doubles, idCoordinateSpaces enumerators, idAnchorPoint enumerators, idBoundingBoxLimits enumerators or Long Integers. / WithMatrix: Transform matrix. Type: Array of 6 Doubles or TransformationMatrix. / ReplacingCurrent: Transform components to consider; providing this optional parameter causes the target's existing transform components to be replaced with new values. Without this parameter, the given matrix is concatenated onto the target's existing transform combining the effect of the two. Type: idMatrixContent enumerator, Array of idMatrixContent enumerators or Long Integer. / ConsideringRulerUnits: If true then a ruler based origin is interpreted using ruler units rather than points. The default value is false. This parameter has no effect unless the reference point is specified relative to a page.</i> |
| transformAgain | <code>variant transformAgain()</code> | <i>Transforms the SplineItem using the last transformation performed on any object. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings.</i> |
| transformAgainIndividually | <code>variant transformAgainIndividually()</code> | <i>Transforms the SplineItem using the last transformation performed on any SplineItem. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings.</i> |
| transformSequenceAgain | <code>variant transformSequenceAgain()</code> | <i>Transforms the SplineItem using the last sequence of transform operations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings.</i> |
| transformSequenceAgainIndividually | <code>variant transformSequenceAgainIndividually()</code> | <i>Transforms the SplineItem using the last sequence of transformations performed on any single object or performed at the same time on any group of objects. Transformations include moving, rotating, shearing, scaling, and flipping. / Return value: Playback errors. Type: Array of Strings.</i> |
| transformValuesOf | <code>{Objects} transformValuesOf(#CoordinateSpaces In)</code> | <i>Get the transformation values of the page item. / Return value: A list of the transformation matrices. / In: The coordinate space to use.</i> |
| absoluteFlip | <code>#Flip absoluteFlip</code> | <i>Indicates whether the SplineItem has been flipped independently of its parent object and, if yes, the direction in which the SplineItem was flipped.</i> |
| absoluteHorizontalScale | <code>double absoluteHorizontalScale</code> | <i>The horizontal scale of the SplineItem relative to its containing object.</i> |
| absoluteRotationAngle | <code>double absoluteRotationAngle</code> | <i>The rotation angle of the SplineItem relative to its containing object. (Range: -360 to 360).</i> |
| absoluteShearAngle | <code>double absoluteShearAngle</code> | <i>The skewing angle of the SplineItem relative to its containing object. (Range: -360 to 360).</i> |
| absoluteVerticalScale | <code>double absoluteVerticalScale</code> | <i>The vertical scale of the SplineItem relative to its containing object.</i> |
| allGraphics | <code>{Objects} *allGraphics</code> | <i>Lists all graphics contained by the SplineItem.</i> |

| SPLINEITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|--|--|
| allowOverrides | <code>boolean allowOverrides</code> | If true, the master page item can be overridden. |
| allPageItems | <code>{Objects} *allPageItems</code> | Lists all page items contained by the SplineItem. |
| anchoredObjectSettings | <code>{AnchoredObjectSetting} *anchoredObjectSettings</code> | Anchored object settings. |
| animationSettings | <code>{AnimationSetting} *animationSettings</code> | The page item animation settings. |
| appliedObjectStyle | <code>{ObjectStyle} appliedObjectStyle</code> | The object style applied to the SplineItem. |
| associatedXMLElement | <code>{XMLElement} *associatedXMLElement</code> | The XML element associated with the SplineItem. |
| bottomLeftCornerOption | <code>#CornerOptions bottomLeftCornerOption</code> | The shape to apply to the bottom left corner of rectangular shapes. |
| bottomLeftCornerRadius | <code>variant bottomLeftCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom left corner of rectangular shapes. Type: Unit (Double or String). |
| bottomRightCornerOption | <code>#CornerOptions bottomRightCornerOption</code> | The shape to apply to the bottom right corner of rectangular shapes. |
| bottomRightCornerRadius | <code>variant bottomRightCornerRadius</code> | The radius in measurement units of the corner effect applied to the bottom right corner of rectangular shapes. Type: Unit (Double or String). |
| buttons | <code>{Buttons} *buttons</code> | A collection of buttons. |
| contentTransparencySettings | <code>{ContentTransparencySetting} *contentTransparencySettings</code> | Transparency settings for the content of the SplineItem. |
| contentType | <code>#ContentType contentType</code> | The type of content that a frame can contain. |
| endCap | <code>#EndCap endCap</code> | The end shape of an open path. |
| endJoin | <code>#EndJoin endJoin</code> | The corner join applied to the SplineItem. |
| epss | <code>{EPSS} *epss</code> | A collection of EPS files. |
| epsTexts | <code>{EPSTexts} *epsTexts</code> | EPSTexts. |
| fillColor | <code>variant fillColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the fill of the SplineItem. . Type: Swatch or String. |
| fillTint | <code>double fillTint</code> | The percent of tint to use in the SplineItem's fill color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| fillTransparencySettings | <code>{FillTransparencySetting} *fillTransparencySettings</code> | Transparency settings for the fill applied to the SplineItem. |
| flip | <code>#Flip flip</code> | The direction in which to flip the printed image. |
| formFields | <code>{FormFields} *formFields</code> | A collection of form fields. |
| gapColor | <code>{Swatch} gapColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the gap of a dashed, dotted, or striped stroke. For information, see stroke type. |
| gapTint | <code>double gapTint</code> | The tint as a percentage of the gap color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| geometricBounds | <code>variant geometricBounds</code> | The bounds of the SplineItem excluding the stroke width, in the format [y1, x1, y2, x2], which give the coordinates of the top-left and bottom-right corners of the bounding box. Type: Array of 4 Units (Doubles or Strings). |
| gradientFillAngle | <code>double gradientFillAngle</code> | The angle of a linear gradient applied to the fill of the SplineItem. (Range: -180 to 180). |
| gradientFillLength | <code>variant gradientFillLength</code> | The length (for a linear gradient) or radius (for a radial gradient) applied to the fill of the SplineItem. Type: Unit (Double or String). |

| SPLINEITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|---|---|
| gradientFillStart | <code>variant gradientFillStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the fill of the SplineItem, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| gradientStrokeAngle | <code>double gradientStrokeAngle</code> | <i>The angle of a linear gradient applied to the stroke of the SplineItem. (Range: -180 to 180).</i> |
| gradientStrokeLength | <code>variant gradientStrokeLength</code> | <i>The length (for a linear gradient) or radius (for a radial gradient) applied to the stroke of the SplineItem. Type: Unit (Double or String).</i> |
| gradientStrokeStart | <code>variant gradientStrokeStart</code> | <i>The starting point (in page coordinates) of a gradient applied to the stroke of the SplineItem, in the format [x, y]. Type: Array of 2 Units (Doubles or Strings).</i> |
| graphicLines | <code>{GraphicLines} *graphicLines</code> | <i>A collection of graphic lines.</i> |
| graphics | <code>{Graphics} *graphics</code> | <i>A collection of imported graphics in any graphic file format (vector, metafile, or bitmap).</i> |
| groups | <code>{Groups} *groups</code> | <i>A collection of groups.</i> |
| horizontalScale | <code>double horizontalScale</code> | <i>The horizontal scaling applied to the SplineItem.</i> |
| id | <code>integer *id</code> | <i>The unique ID of the SplineItem.</i> |
| images | <code>{Images} *images</code> | <i>A collection of bitmap images in any bitmap file format (including TIFF, JPEG, or GIF).</i> |
| importedPages | <code>{ImportedPages} *importedPages</code> | <i>Imported InDesign pages.</i> |
| index | <code>integer *index</code> | <i>The index of the SplineItem within its containing object.</i> |
| itemLayer | <code>{Layer} itemLayer</code> | <i>The layer that the SplineItem is on.</i> |
| label | <code>string label</code> | <i>A property that can be set to any string.</i> |
| leftLineEnd | <code>#ArrowHead leftLineEnd</code> | <i>The arrowhead applied to the start of the path.</i> |
| localDisplaySetting | <code>#DisplaySettingOptions localDisplaySetting</code> | <i>Display performance options for the SplineItem.</i> |
| locked | <code>boolean locked</code> | <i>If true, the SplineItem is locked.</i> |
| lockState | <code>#LockStateValues *lockState</code> | <i>The lock state.</i> |
| mediaItems | <code>{MediaItems} *mediaItems</code> | <i>The media items collection.</i> |
| miterLimit | <code>double miterLimit</code> | <i>The limit of the ratio of stroke width to miter length before a miter (pointed) join becomes a bevel (squared-off) join. Type: Double (1 - 500).</i> |
| movies | <code>{Movies} *movies</code> | <i>A collection of movies.</i> |
| multiStateObjects | <code>{MultiStateObjects} *multiStateObjects</code> | <i>A collection of multi-state objects.</i> |
| name | <code>string name</code> | <i>The name of the SplineItem.</i> |
| nonprinting | <code>boolean nonprinting</code> | <i>If true, the SplineItem does not print.</i> |
| ovals | <code>{Ovals} *ovals</code> | <i>A collection of ellipses.</i> |
| overprintFill | <code>boolean overprintFill</code> | <i>If true, the SplineItem's fill color overprints any underlying objects. If false, the fill color knocks out the underlying colors.</i> |
| overprintGap | <code>boolean overprintGap</code> | <i>If true, the gap color overprints any underlying colors. If false, the gap color knocks out the underlying colors.</i> |
| overprintStroke | <code>boolean overprintStroke</code> | <i>If true, the SplineItem's stroke color overprints any underlying objects. If false, the stroke color knocks out the underlying colors.</i> |

| SPLINEITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|--|
| overridden | <code>boolean *overridden</code> | If true, the object originated on a master spread and was overridden. If false, the object either originated on a master spread and was not overridden, or the object did not originate on a master page. |
| overriddenMasterPageItem | <code>variant *overriddenMasterPageItem</code> | An object that originated on a master page and has been overridden. Type: PageItem, Guide, Graphic, Movie or Sound. |
| pageItems | <code>{PageItems} *pageItems</code> | The page items collection, which can be used to process all page items in a container (such as a document, page, or group), regardless of type. |
| parent | <code>variant *parent</code> | The parent of the SplineItem (a Snippet, Spread, MasterSpread, Polygon, GraphicLine, Rectangle, Oval, SplineItem, Group, State, Character or PlaceGun). Type: Snippet, Spread, MasterSpread, Polygon, GraphicLine, Rectangle, Oval, SplineItem, Group, State, Character or PlaceGun. |
| parentPage | <code>{Page} *parentPage</code> | The page on which this page item appears. |
| paths | <code>{Paths} *paths</code> | A collection of paths. |
| pdfs | <code>{PDFs} *pdfs</code> | A collection of PDF files. |
| picts | <code>{PICTs} *picts</code> | A collection of PICT graphics. |
| polygons | <code>{Polygons} *polygons</code> | A collection of polygons. |
| preferences | <code>{Preferences} *preferences</code> | A collection of preferences objects. |
| rectangles | <code>{Rectangles} *rectangles</code> | A collection of rectangles. |
| rightLineEnd | <code>#ArrowHead rightLineEnd</code> | The arrowhead applied to the end of the path. |
| rotationAngle | <code>double rotationAngle</code> | The rotation angle of the SplineItem. (Range: -360 to 360). |
| shearAngle | <code>double shearAngle</code> | The skewing angle applied to the SplineItem. (Range: -360 to 360). |
| sounds | <code>{Sounds} *sounds</code> | A collection of sound clips. |
| splineItems | <code>{SplineItems} *splineItems</code> | The spline items collection. |
| strokeAlignment | <code>#StrokeAlignment strokeAlignment</code> | The stroke alignment applied to the SplineItem. |
| strokeColor | <code>variant strokeColor</code> | The swatch (color, gradient, tint, or mixed ink) applied to the stroke of the SplineItem. Type: Swatch or String. |
| strokeCornerAdjustment | <code>#StrokeCornerAdjustment strokeCornerAdjustment</code> | The corner adjustment applied to the SplineItem. |
| strokeDashAndGap | <code>variant strokeDashAndGap</code> | The dash and gap measurements that define the pattern of a custom dashed line. Define up to six values (in points) in the format [dash1, gap1, dash2, gap2, dash3, gap3]. Type: Array of Units (Doubles or Strings). |
| strokeTint | <code>double strokeTint</code> | The percent of tint to use in object's stroke color. (To specify a tint percent, use a number in the range of 0 to 100; to use the inherited or overridden value, use -1.). |
| strokeTransparencySettings | <code>{StrokeTransparencySetting} *strokeTransparencySettings</code> | Transparency settings for the stroke. |
| strokeType | <code>variant strokeType</code> | The name of the stroke style to apply. Type: StrokeStyle or String. |
| strokeWeight | <code>variant strokeWeight</code> | The weight (in points) to apply to the SplineItem's stroke. Type: Unit (Double or String). |
| textFrames | <code>{TextFrames} *textFrames</code> | A collection of text frames. |
| textPaths | <code>{TextPaths} *textPaths</code> | A collection of text paths. |
| textWrapPreferences | <code>{TextWrapPreference} *textWrapPreferences</code> | The text wrap preference properties that define the default formatting for wrapping text around objects. |

| SPLINEITEM CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------------|--|--|
| <code>timingSettings</code> | <code>{TimingSetting} *timingSettings</code> | <i>The object timing settings.</i> |
| <code>topLeftCornerOption</code> | <code>#CornerOptions topLeftCornerOption</code> | <i>The shape to be applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Note: corner option differs from end join in which you can set a radius for a corner option, whereas the rounded or beveled effect of an end join depends on the stroke weight.</i> |
| <code>topLeftCornerRadius</code> | <code>variant topLeftCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top left corner of rectangular shapes and all corners of non-rectangular shapes. Type: Unit (Double or String).</i> |
| <code>topRightCornerOption</code> | <code>#CornerOptions topRightCornerOption</code> | <i>The shape to apply to the top right corner of rectangular shapes.</i> |
| <code>topRightCornerRadius</code> | <code>variant topRightCornerRadius</code> | <i>The radius in measurement units of the corner effect applied to the top right corner of rectangular shapes. Type: Unit (Double or String).</i> |
| <code>transparencySettings</code> | <code>{TransparencySetting} *transparencySettings</code> | <i>Transparency settings.</i> |
| <code>verticalScale</code> | <code>double verticalScale</code> | <i>The vertical scaling applied to the SplineItem.</i> |
| <code>visible</code> | <code>boolean visible</code> | <i>If true, the SplineItem is visible.</i> |
| <code>visibleBounds</code> | <code>variant visibleBounds</code> | <i>The bounds of the SplineItem including the stroke width, in the format [y1, x1, y2, x2], which give the coordinates of the top-left and bottom-right corners of the bounding box. Type: Array of 4 Units (Doubles or Strings).</i> |
| <code>wmfs</code> | <code>{WMFs} *wmfs</code> | <i>A collection of WMF graphics.</i> |

SplineItemEvents

Object available in ID 7

| SPLINEITEMEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|--|---|
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a SplineItem is placed. This event bubbles. This event is not cancelable.</i> |
| <code>beforePlace</code> | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a SplineItem is placed. This event bubbles. This event is cancelable.</i> |

SplineItems

Object available in ID 7

| SPLINEITEMS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|--|---|
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | <i>Returns any object in the collection. / Return value: The object.</i> |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | <i>Returns the first object in the collection. / Return value: The first object.</i> |
| <code>itemByID</code> | <code>IDispatch itemByID(integer Id)</code> | <i>Returns the object with the specified ID. / Return value: The object. / ID: The ID.</i> |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | <i>Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String.</i> |

| SPLINEITEMS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|--|
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| <code>previousItem</code> | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| <code>count</code> | <code>integer *count</code> | Displays the number of elements in the SplineItem. |
| <code>item</code> | <code>{SplineItem} *item(variant Index)</code> | Returns the object with the specified index or name. Type: Long Integer or String. |

StoryEvents

Object available in ID 7

| STORYEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------|--|---|
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | Dispatched after a Story is placed. This event bubbles. This event is not cancelable. |
| <code>beforePlace</code> | <code>const beforePlace(beforePlace)</code> | Dispatched before a Story is placed. This event bubbles. This event is cancelable. |

StoryGridDataInformation

Object available in ID 7

| STORYGRIDDATAINFORMATION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|---|---|
| <code>appliedFont</code> | <code>variant appliedFont</code> | The font applied to the StoryGridDataInformation, specified as either a font object or the name of font family. Type: Font or String. |
| <code>characterAki</code> | <code>double characterAki</code> | The amount of white space between characters. |
| <code>characterAlignment</code> | <code>#CharacterAlignment characterAlignment</code> | The alignment of small characters to the largest character in the line. |
| <code>characterCountLocation</code> | <code>#CharacterCountLocation characterCountLocation</code> | The character count location. Note: Valid when show character count is true. |
| <code>characterCountSize</code> | <code>double characterCountSize</code> | The character size for the character count display. |
| <code>fontStyle</code> | <code>string fontStyle</code> | The name of the font style. |
| <code>gridAlignment</code> | <code>#GridAlignment gridAlignment</code> | The alignment to the frame grid or baseline grid. |
| <code>gridView</code> | <code>#GridViewSettings gridView</code> | The grid view setting. |
| <code>horizontalScale</code> | <code>double horizontalScale</code> | The horizontal scaling applied to the StoryGridDataInformation. |
| <code>lineAki</code> | <code>double lineAki</code> | The amount of white space between lines. |
| <code>lineAlignment</code> | <code>#LineAlignment lineAlignment</code> | The line justification. |

| STORYGRIDDATAINFORMATION CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|-----------------------------------|--|
| parent | <code>variant •parent</code> | The parent of the StoryGridDataInformation (a Application or Document). Type: Application or Document. |
| pointSize | <code>variant pointSize</code> | The text size. Type: Unit (Double or String). |
| verticalScale | <code>double verticalScale</code> | The vertical scaling applied to the StoryGridDataInformation. |

StoryWindowEvents

Object available in ID 7

| STORYWINDOWEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------------|--|---|
| afterActivate | <code>const afterActivate(afterActivate)</code> | Dispatched after the StoryWindow becomes active. This event bubbles. This event is not cancelable. |
| afterAttributeChanged | <code>const afterAttributeChanged(afterAttributeChanged)</code> | Dispatched when the value of a property changes on this StoryWindow. This event bubbles. This event is not cancelable. |
| afterClose | <code>const afterClose(afterClose)</code> | Dispatched when a StoryWindow is closing. Since the close has been committed, it can no longer be canceled. This event bubbles. This event is not cancelable. |
| afterOpen | <code>const afterOpen(afterOpen)</code> | Dispatched after a StoryWindow is opened. This event bubbles. This event is not cancelable. |
| afterSelectionAttributeChanged | <code>const afterSelectionAttributeChanged(afterSelectionAttributeChanged)</code> | Dispatched after an attribute on the active selection changes. This event bubbles. This event is not cancelable. |
| afterSelectionChanged | <code>const afterSelectionChanged(afterSelectionChanged)</code> | Dispatched after the active selection changes. This event bubbles. This event is not cancelable. |
| beforeClose | <code>const beforeClose(beforeClose)</code> | Dispatched before a StoryWindow is closed. This event bubbles. This event is cancelable. |
| beforeDeactivate | <code>const beforeDeactivate(beforeDeactivate)</code> | Dispatched before the StoryWindow becomes inactive. This event bubbles. This event is not cancelable. |

SubmenuEvents

Object available in ID 7

| SUBMENUEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|---|
| beforeDisplay | <code>const beforeDisplay(beforeDisplay)</code> | Dispatched before the Submenu is displayed. This event does not bubble. This event is not cancelable. |

TextFrameEvents

Object available in ID 7

| TEXTFRAMEEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------------|--|---|
| afterPlace | <code>const afterPlace(afterPlace)</code> | Dispatched after a TextFrame is placed. This event bubbles. This event is not cancelable. |
| beforePlace | <code>const beforePlace(beforePlace)</code> | Dispatched before a TextFrame is placed. This event bubbles. This event is cancelable. |

TextPathEvents

Object available in ID 7

| TEXTPATHEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|--|
| afterPlace | <code>const afterPlace(afterPlace)</code> | Dispatched after a TextPath is placed. This event bubbles. This event is not cancelable. |
| beforePlace | <code>const beforePlace(beforePlace)</code> | Dispatched before a TextPath is placed. This event bubbles. This event is cancelable. |

TimingGroup

Object available in ID 7

| TIMINGGROUP CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------|---|--|
| delete | <code>void delete()</code> | Deletes the TimingGroup. |
| move | <code>{TimingGroup} move(#LocationOptions To, variant Reference)</code> | Moves the timing group or target to the specified location. / Return value: The moved element. / To: The location in relation to the reference object or within the containing object. / Reference: The reference object. Note: Required when the to parameter specifies before or after. Type: TimingGroup, TimingTarget or TimingList. |
| unlink | <code>void unlink()</code> | Unlink all targets in the group into separate groups in this timing list. |
| index | <code>integer •index</code> | The index of the TimingGroup within its containing object. |
| parent | <code>{TimingList} •parent</code> | The parent of the TimingGroup (a TimingList). |
| placement | <code>integer placement</code> | The placement of the timing group in the timing list. |
| plays | <code>integer plays</code> | The number of times this timing group plays. Type: Long Integer (1 - 100). |
| playsLoop | <code>boolean playsLoop</code> | Set to true if timing group loops. |
| timingTargets | <code>{TimingTargets} •timingTargets</code> | A collection of timing target. |

TimingGroups

Object available in ID 7

| TIMINGGROUPS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|--|
| <code>add</code> | <code>{TimingGroup} add(variant DynamicTarget, double DelaySeconds)</code> | Adds a new timing group. / Return value: The timing group added. / DynamicTarget: A page item target that is an animation, media, or mso. Type: PageItem, Graphic, Behavior or MediaItem. / DelaySeconds: The time delay in seconds for a target. |
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | Returns any object in the collection. / Return value: The object. |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | Returns the first object in the collection. / Return value: The first object. |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| <code>previousItem</code> | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| <code>count</code> | <code>integer *count</code> | Displays the number of elements in the TimingGroup. |
| <code>item</code> | <code>{TimingGroup} *item(variant Index)</code> | Returns the object with the specified index or name. Type: Long Integer or String. |

TimingList

Object available in ID 7

| TIMINGLIST CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|--|---|
| <code>delete</code> | <code>void delete()</code> | Deletes the TimingList. |
| <code>index</code> | <code>integer *index</code> | The index of the TimingList within its containing object. |
| <code>parent</code> | <code>{TimingSetting} *parent</code> | The parent of the TimingList (a TimingSetting). |
| <code>timingGroups</code> | <code>{TimingGroups} *timingGroups</code> | A collection of timing data objects. |
| <code>triggerEvent</code> | <code>#DynamicTriggerEvents *triggerEvent</code> | trigger event. |

TimingLists

Object available in ID 7

| TIMINGLISTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------|--|--|
| add | {TimingList} add(#DynamicTriggerEvents TriggerEvent) | Adds a new event-triggered timing list object. / Return value: The timing list to add. / TriggerEvent: Trigger Event. |
| anyItem | IDispatch anyItem() | Returns any object in the collection. / Return value: The object. |
| firstItem | IDispatch firstItem() | Returns the first object in the collection. / Return value: The first object. |
| itemByRange | {Objects} itemByRange(variant From, variant To) | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| lastItem | IDispatch lastItem() | Returns the last object in the collection. / Return value: The last object. |
| middleItem | IDispatch middleItem() | Returns the middle object in the collection. / Return value: The middle object. |
| nextItem | IDispatch nextItem(IDispatch Obj) | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| previousItem | IDispatch previousItem(IDispatch Obj) | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| count | integer *count | Displays the number of elements in the TimingList. |
| item | {TimingList} *item(variant Index) | Returns the object with the specified index or name. Type: Long Integer or String. |

TimingSetting

Object available in ID 7

| TIMINGSETTING CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|-------------------------------------|---|
| parent | variant *parent | The parent of the TimingSetting (a PageItem, FormField, MultiStateObject, Button, MediaItem, Sound, Movie, EPSText, SplineItem, Polygon, GraphicLine, Rectangle, Oval, Graphic, ImportedPage, PICT, WMF, PDF, EPS, Image, Group, TextFrame, MasterSpread or Spread). Type: PageItem, FormField, MultiStateObject, Button, MediaItem, Sound, Movie, EPSText, SplineItem, Polygon, GraphicLine, Rectangle, Oval, Graphic, ImportedPage, PICT, WMF, PDF, EPS, Image, Group, TextFrame, MasterSpread or Spread. |
| timingLists | {TimingLists} *timingLists | A collection of timing lists. |
| unassignedDynamicTargets | {Objects} *unassignedDynamicTargets | Dynamic targets on the spread that are not assigned. |

TimingTarget

Object available in ID 7

| TIMINGTARGET CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------------|--|--|
| <code>delete</code> | <code>void delete()</code> | Deletes the TimingTarget. |
| <code>move</code> | <code>{TimingTarget} move(#LocationOptions To, variant Reference)</code> | Moves the timing group or target to the specified location. / Return value: The moved element. / To: The location in relation to the reference object or within the containing object. / Reference: The reference object. Note: Required when the to parameter specifies before or after. . Type: TimingGroup, TimingTarget or TimingList. |
| <code>unlink</code> | <code>void unlink()</code> | Unlink target from this group and append to the end of this timing list. |
| <code>delaySeconds</code> | <code>double delaySeconds</code> | The time delay in seconds for a single target or a group of targets after the previous group has finished. Type: Double (0 - 60). |
| <code>dynamicTarget</code> | <code>variant dynamicTarget</code> | A page item target that is an animation, media, or mso. Type: PageItem, Graphic, Behavior or MediaItem. |
| <code>index</code> | <code>integer *index</code> | The index of the TimingTarget within its containing object. |
| <code>parent</code> | <code>{TimingGroup} *parent</code> | The parent of the TimingTarget (a TimingGroup). |
| <code>reverseAnimation</code> | <code>boolean reverseAnimation</code> | Set to true if reversing animation on rolloff. Valid only for self rolloff trigger event. |

TimingTargets

Object available in ID 7

| TIMINGTARGETS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|---|--|
| <code>add</code> | <code>{TimingTarget} add(variant DynamicTarget, double DelaySeconds)</code> | Adds a new target item. / Return value: The target item added. / DynamicTarget: A page item target that is an animation, media, or mso. Type: PageItem, Graphic, Behavior or MediaItem. / DelaySeconds: The time delay in seconds for a target. |
| <code>anyItem</code> | <code>IDispatch anyItem()</code> | Returns any object in the collection. / Return value: The object. |
| <code>firstItem</code> | <code>IDispatch firstItem()</code> | Returns the first object in the collection. / Return value: The first object. |
| <code>itemByRange</code> | <code>{Objects} itemByRange(variant From, variant To)</code> | Returns the objects within the specified range. / Return value: The objects. / From: The object, index, or name at the beginning of the range. Type: Object, Long Integer or String. / To: The object, index, or name at the end of the range. Type: Object, Long Integer or String. |
| <code>lastItem</code> | <code>IDispatch lastItem()</code> | Returns the last object in the collection. / Return value: The last object. |
| <code>middleItem</code> | <code>IDispatch middleItem()</code> | Returns the middle object in the collection. / Return value: The middle object. |
| <code>nextItem</code> | <code>IDispatch nextItem(IDispatch Obj)</code> | Returns the object whose index follows the specified object in the collection. / Return value: The next object. / Obj: The object whose index comes before the desired object. |
| <code>previousItem</code> | <code>IDispatch previousItem(IDispatch Obj)</code> | Returns the object with the index previous to the specified index. / Return value: The previous object. / Obj: The index of the object that follows the desired object. |
| <code>count</code> | <code>integer *count</code> | Displays the number of elements in the TimingTarget. |

| TIMINGTARGETS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------|---|--|
| item | {TimingTarget} •item(<i>variant</i> Index) | Returns the object with the specified index or name. Type: Long Integer or String. |

ToolBox

Object available in ID 7

| TOOLBOX CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|-----------------------------|---|
| currentTool | #UITools currentTool | The currently active tool. |
| currentToolHint | string •currentToolHint | The currently active tool's hint. |
| currentToolIconFile | string •currentToolIconFile | The currently active tool's icon resource file. |
| currentToolName | string •currentToolName | The currently active tool's name. |
| parent | {Application} •parent | The parent of the ToolBox (a Application). |

ToolBoxEvents

Object available in ID 7

| TOOLBOXEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------|---|--|
| afterAttributeChanged | const afterAttributeChanged(afterAttributeChanged) | Dispatched when the value of a property changes on this ToolBox. This event bubbles. This event is not cancelable. |

TrackChangesPreference

Object available in ID 7

| TRACKCHANGESPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--------------------------------------|--|---|
| addedBackgroundColorChoice | #ChangeBackgroundColorChoices addedBackgroundColorChoice | The background color option for added text. |
| addedTextColorChoice | #ChangeTextColorChoices addedTextColorChoice | The color option for added text. |
| backgroundColorForAddedText | <i>variant</i> backgroundColorForAddedText | The background color for added text, specified as an InCopy UI color. Note: Valid only when added background color choice is change background uses change pref color. Type: Array of 3 Doubles (0 - 255) or idInCopyUIColors enumerator. |
| backgroundColorForDeletedText | <i>variant</i> backgroundColorForDeletedText | The background color for deleted text, specified as an InCopy UI color. Note: Valid only when deleted background color choice is change background uses change pref color. Type: Array of 3 Doubles (0 - 255) or idInCopyUIColors enumerator. |
| backgroundColorForMovedText | <i>variant</i> backgroundColorForMovedText | The background color for moved text, specified as an InCopy UI color. Note: Valid only when moved background color choice is change background uses change pref color. Type: Array of 3 Doubles (0 - 255) or idInCopyUIColors enumerator. |

| TRACKCHANGESPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|------------------------------------|--|---|
| changeBarColor | variant changeBarColor | The change bar color, specified as an InCopy UI color. Type: Array of 3 Doubles (0 - 255) or idInCopyUIColors enumerator. |
| deletedBackgroundColorChoice | #ChangeBackgroundColorChoices deletedBackgroundColorChoice | The background color option for deleted text. |
| deletedTextColorChoice | #ChangeTextColorChoices deletedTextColorChoice | The color option for deleted text. |
| locationForChangeBar | #ChangebarLocations locationForChangeBar | The change bar location. |
| markingForAddedText | #ChangeMarkings markingForAddedText | The marking that identifies added text. |
| markingForDeletedText | #ChangeMarkings markingForDeletedText | The marking that identifies deleted text. |
| markingForMovedText | #ChangeMarkings markingForMovedText | The marking that identifies moved text. |
| movedBackgroundColorChoice | #ChangeBackgroundColorChoices movedBackgroundColorChoice | The background color option for moved text. |
| movedTextColorChoice | #ChangeTextColorChoices movedTextColorChoice | The color option for moved text. |
| parent | {Application} *parent | The parent of the TrackChangesPreference (a Application). |
| preventDuplicateColor | boolean preventDuplicateColor | If true, prevent duplicate user color for tracked changes background color. |
| showAddedText | boolean showAddedText | If true, displays added text. |
| showChangeBars | boolean showChangeBars | If true, displays change bars.. |
| showDeletedText | boolean showDeletedText | If true, displays deleted text. |
| showMovedText | boolean showMovedText | If true, displays moved text. |
| spellCheckDeletedText | boolean spellCheckDeletedText | If true, includes deleted text when using the Spell Check command. |
| textColorForAddedText | variant textColorForAddedText | The color for added text, specified as an InCopy UI color. Note: Valid only when added text color choice is change uses change pref color. Type: Array of 3 Doubles (0 - 255) or idInCopyUIColors enumerator. |
| textColorForDeletedText | variant textColorForDeletedText | The color for deleted text, specified as an InCopy UI color. Note: Valid only when deleted text color choice is change uses change pref color. Type: Array of 3 Doubles (0 - 255) or idInCopyUIColors enumerator. |
| textColorForMovedText | variant textColorForMovedText | The color for moved text, specified as an InCopy UI color. Note: Valid only when moved text color choice is change uses change pref color. Type: Array of 3 Doubles (0 - 255) or idInCopyUIColors enumerator. |

WatermarkPreference

Object available in ID 7

| WATERMARKPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---------------------------------|-----------------------------|---|
| parent | variant *parent | The parent of the WatermarkPreference (a Document or Application). Type: Document or Application. |
| watermarkDoPrint | boolean watermarkDoPrint | Watermark do print for a document. |
| watermarkDrawInBack | boolean watermarkDrawInBack | Watermark draw in back for a document. |
| watermarkFontColor | variant watermarkFontColor | Watermark font color for a document. Type: Array of 3 Doubles (0 - 255) or idUIColors enumerator. |
| watermarkFontFamily | string watermarkFontFamily | Watermark font family display name. |

| WATERMARKPREFERENCE CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|--|---|--|
| <code>watermarkFontSize</code> | <code>integer watermarkFontSize</code> | Watermark font point size for a document. |
| <code>watermarkFontStyle</code> | <code>string watermarkFontStyle</code> | Watermark font style name. |
| <code>watermarkHorizontalOffset</code> | <code>variant watermarkHorizontalOffset</code> | Watermark horizontal offset for a document. Type: Unit (Double or String). |
| <code>watermarkHorizontalPosition</code> | <code>#WatermarkHorizontalPositionEnum watermarkHorizontalPosition</code> | Watermark horizontal position enum for a document. |
| <code>watermarkOpacity</code> | <code>integer watermarkOpacity</code> | Watermark opacity (as a percentage). (Range: 0 to 100). |
| <code>watermarkRotation</code> | <code>integer watermarkRotation</code> | Watermark rotation for a document. |
| <code>watermarkText</code> | <code>string watermarkText</code> | Watermark text for a document. |
| <code>watermarkVerticalOffset</code> | <code>variant watermarkVerticalOffset</code> | Watermark vertical offset for a document. Type: Unit (Double or String). |
| <code>watermarkVerticalPosition</code> | <code>#WatermarkVerticalPositionEnum watermarkVerticalPosition</code> | Watermark vertical position enum for a document. |
| <code>watermarkVisibility</code> | <code>boolean watermarkVisibility</code> | Watermark visibility for a document. |

WindowEvents

Object available in ID 7

| WINDOWEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|---|--|--|
| <code>afterActivate</code> | <code>const afterActivate(afterActivate)</code> | Dispatched after the Window becomes active. This event bubbles. This event is not cancelable. |
| <code>afterAttributeChanged</code> | <code>const afterAttributeChanged(afterAttributeChanged)</code> | Dispatched when the value of a property changes on this Window. This event bubbles. This event is not cancelable. |
| <code>afterClose</code> | <code>const afterClose(afterClose)</code> | Dispatched when a Window is closing. Since the close has been committed, it can no longer be canceled. This event bubbles. This event is not cancelable. |
| <code>afterOpen</code> | <code>const afterOpen(afterOpen)</code> | Dispatched after a Window is opened. This event bubbles. This event is not cancelable. |
| <code>afterSelectionAttributeChanged</code> | <code>const afterSelectionAttributeChanged(afterSelectionAttributeChanged)</code> | Dispatched after an attribute on the active selection changes. This event bubbles. This event is not cancelable. |
| <code>afterSelectionChanged</code> | <code>const afterSelectionChanged(afterSelectionChanged)</code> | Dispatched after the active selection changes. This event bubbles. This event is not cancelable. |
| <code>beforeClose</code> | <code>const beforeClose(beforeClose)</code> | Dispatched before a Window is closed. This event bubbles. This event is cancelable. |
| <code>beforeDeactivate</code> | <code>const beforeDeactivate(beforeDeactivate)</code> | Dispatched before the Window becomes inactive. This event bubbles. This event is not cancelable. |

WMFEvents

Object available in ID 7

| WMFEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-------------------------|--|---|
| <code>afterPlace</code> | <code>const afterPlace(afterPlace)</code> | Dispatched after a WMF is placed. This event bubbles. This event is not cancelable. |

| WMFEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|-----------------------|--|---|
| beforePlace | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a WMF is placed. This event bubbles. This event is cancelable.</i> |

XmlStoryEvents

Object available in ID 7

| XMLSTORYEVENTS CS5 MEMBERS | PROTOTYPE | DESCRIPTION |
|----------------------------|--|---|
| afterPlace | <code>const afterPlace(afterPlace)</code> | <i>Dispatched after a XmlStory is placed. This event bubbles. This event is not cancelable.</i> |
| beforePlace | <code>const beforePlace(beforePlace)</code> | <i>Dispatched before a XmlStory is placed. This event bubbles. This event is cancelable.</i> |