

ADOBE® ILLUSTRATOR® CS3

VBSCRIPT REFERENCE

© 2007 Adobe Systems Incorporated. All rights reserved.

Adobe® Illustrator® CS3 VBScript Reference for Windows® and Mac OS®.

NOTICE: All information contained herein is the property of Adobe Systems Incorporated. No part of this publication (whether in hardcopy or electronic form) may be reproduced or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Adobe Systems Incorporated. The software described in this document is furnished under license and may only be used or copied in accordance with the terms of such license.

This publication and the information herein is furnished AS IS, is subject to change without notice, and should not be construed as a commitment by Adobe Systems Incorporated. Adobe Systems Incorporated assumes no responsibility or liability for any errors or inaccuracies, makes no warranty of any kind (express, implied, or statutory) with respect to this publication, and expressly disclaims any and all warranties of merchantability, fitness for particular purposes, and noninfringement of third party rights.

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

Adobe, the Adobe logo, Acrobat, Photoshop, and Illustrator are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Apple, Mac, Macintosh, and Mac OS are trademarks of Apple Computer, Inc., registered in the United States and other countries. Microsoft, and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and other countries. JavaScript and all Java-related marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries. UNIX is a registered trademark of The Open Group.

All other trademarks are the property of their respective owners.

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

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

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

Contents

1	VBScript Object Reference	10
	Application	11
	Application properties	11
	Application methods	12
	Brush	17
	Brush properties	17
	Brush methods	17
	Brushes	18
	Brushes properties	18
	Brushes methods	18
	CharacterAttributes	19
	CharacterAttributes properties	19
	Characters	23
	Characters properties	23
	Characters methods	23
	CharacterStyle	24
	CharacterStyle properties	24
	CharacterStyle methods	24
	CharacterStyles	25
	CharacterStyles properties	25
	CharacterStyles methods	25
	CMYKColor	27
	CMYKColor properties	27
	CompoundPathItem	28
	CompoundPathItem properties	28
	CompoundPathItem methods	29
	CompoundPathItems	32
	CompoundPathItems properties	32
	CompoundPathItem methods	32
	DataSet	33
	DataSet properties	33
	DataSet methods	33
	DataSets	34
	DataSets properties	34
	DataSets methods	34
	Document	36
	Document properties	36
	Document methods	40
	Documents	43
	Documents properties	43
	Documents methods	43
	DocumentPreset	44
	DocumentPreset properties	44
	EPSSaveOptions	45
	EPSSaveOptions properties	45

ExportOptionsAutoCAD.....	47
ExportOptionsAutoCAD properties	47
ExportOptionsFlash	48
ExportOptionsFlash properties	48
ExportOptionsGIF	50
ExportOptionsGIF properties	50
ExportOptionsJPEG.....	52
ExportOptionsJPEG	52
ExportOptionsPhotoshop	54
ExportOptionsPhotoshop properties	54
ExportOptionsPNG8	55
ExportOptionsPNG8 properties	55
ExportOptionsPNG24.....	57
ExportOptionsPNG24 properties	57
ExportOptionsSVG	58
ExportOptionsSVG properties	58
FreeHandFileOptions.....	60
FreeHandFileOptions properties	60
Gradient	61
Gradient properties	61
Gradient methods.....	61
GradientColor	63
GradientColor properties	63
Gradients	64
Gradients properties	64
Gradients methods.....	64
GradientStop.....	65
GradientStop properties.....	65
GradientStop methods	65
GradientStops.....	66
GradientStops properties.....	66
GradientStops methods	66
GraphicStyle	68
GraphicStyle properties	68
GraphicStyle methods.....	68
GraphicStyles	69
GraphicStyles properties	69
GraphicStyles methods.....	69
GraphItem	70
GraphItem properties.....	70
GraphItem methods.....	71
GraphItems	73
GraphItems properties	73
GraphItems methods.....	73
GrayColor.....	74
GrayColor properties	74
GroupItem.....	75
GroupItem properties.....	75
GroupItem methods	77
GroupItems.....	79
GroupItems properties.....	79

GroupItems methods	79
IllustratorSaveOptions.....	80
IllustratorSaveOptions properties.....	80
ImageCaptureOptions.....	81
ImageCaptureOptions properties.....	81
Ink.....	82
Ink properties	82
InkInfo.....	82
InkInfo properties.....	82
InsertionPoint	84
InsertionPoints	84
InsertionPoints properties	84
InsertionPoints methods.....	84
LabColor	85
LabColor properties	85
Layer.....	86
Layer properties.....	86
Layer methods	87
Layers.....	89
Layers properties.....	89
Layers methods.....	89
LegacyTextItem.....	90
LegacyTextItem properties	90
LegacyTextItem methods	91
LegacyTextItems.....	93
LegacyTextItems properties.....	93
LegacyTextItems methods	93
Lines	94
Lines properties	94
Lines methods	94
Matrix.....	95
Matrix properties.....	95
MeshItem.....	96
MeshItem properties	96
MeshItem methods	97
MeshItems.....	99
MeshItems properties.....	99
MeshItems methods	99
NoColor	101
NoColor properties.....	101
OpenOptions	102
OpenOptions properties	102
OpenOptionsAutoCAD.....	103
OpenOptionsAutoCAD properties	103
OpenOptionsPhotoshop	104
OpenOptionsPhotoshop properties	104
PageItems.....	105
PageItems properties	105
PageItems methods	105
Paper	107
Paper properties.....	107

PaperInfo	107
PaperInfo properties	107
ParagraphAttributes	109
ParagraphAttributes properties	109
Paragraphs	113
Paragraphs properties	113
Paragraphs methods	113
ParagraphStyle	114
ParagraphStyle properties	114
ParagraphStyle methods	114
ParagraphStyles	116
ParagraphStyles properties	116
ParagraphStyles methods	116
PathItem	117
PathItem properties	117
PathItem methods	119
PathItems	122
PathItems properties	122
PathItems methods	122
PathPoint	124
PathPoint properties	124
PathPoint methods	124
PathPoints	126
PathPoints properties	126
PathPoints methods	126
Pattern	127
Pattern properties	127
Pattern methods	127
PatternColor	128
PatternColor properties	128
Patterns	130
Patterns properties	130
Patterns methods	130
PDFFileOptions	131
PDFFileOptions properties	131
PDFSaveOptions	132
PDFSaveOptions properties	132
PhotoshopFileOptions	136
PhotoshopFileOptions properties	136
PlacedItem	137
PlacedItem properties	137
PlacedItem methods	138
PlacedItems	141
PlacedItems properties	141
PlacedItems methods	141
PluginItem	142
PluginItem properties	142
PluginItem methods	143
PluginItems	146
PluginItems properties	146
PluginItems methods	146

PPDFile	147
PPDFile properties	147
PPDFileInfo	147
PPDFileInfo properties	147
Preferences	149
Preferences properties	149
Preferences methods	149
PrintColorManagementOptions	151
PrintColorManagementOptions properties	151
PrintColorSeparationOptions	152
PrintColorSeparationOptions properties	152
PrintCoordinateOptions	154
PrintCoordinateOptions properties	154
Printer	156
Printer properties	156
PrinterInfo	156
PrinterInfo properties	156
PrintFlattenerOptions	159
PrintFlattenerOptions properties	159
PrintFontOptions	161
PrintFontOptions properties	161
PrintJobOptions	162
PrintJobOptions properties	162
PrintOptions	164
PrintOptions properties	164
PrintPageMarksOptions	166
PrintPageMarksOptions properties	166
PrintPaperOptions	167
PrintPaperOptions properties	167
PrintPostScriptOptions	168
PrintPostScriptOptions properties	168
RasterItem	170
RasterItem properties	170
RasterItem methods	172
RasterItems	174
RasterItems properties	174
RasterItems methods	174
RGBColor	175
RGBColor properties	175
Screen	176
Screen properties	176
ScreenInfo	176
ScreenInfo properties	176
ScreenSpotFunction	178
ScreenSpotFunction properties	178
Spot	179
Spot properties	179
Spot methods	179
SpotColor	180
SpotColor properties	180
Spots	181

Spots properties	181
Spots methods	181
Story	182
Story properties	182
Stories	184
Stories properties	184
Stories methods	184
Swatch	185
Swatch properties	185
Swatch methods	185
Swatches	186
Swatches properties	186
Swatches methods	186
Symbol	187
Symbol properties	187
Symbol methods	187
Symbols	188
Symbols properties	188
Symbols methods	188
SymbolItem	189
SymbolItem properties	189
SymbolItem methods	190
SymbolItems	193
SymbolItems properties	193
SymbolItems methods	193
TabStopInfo	194
TabStopInfo properties	194
Tag	195
Tag properties	195
Tag methods	195
Tags	196
Tags properties	196
Tags methods	196
TextFont	197
TextFont properties	197
TextFonts	199
TextFonts properties	199
TextFonts methods	199
TextFrame	200
TextFrame properties	200
TextFrame methods	202
TextFrames	205
TextFrames properties	205
TextFrames methods	205
TextPath	207
TextPath properties	207
TextPath methods	208
TextRange	209
TextRange properties	209
TextRange methods	210
TextRanges	212

TextRanges properties	212
TextRanges methods	212
TracingObject	213
TracingObject properties	213
TracingObject methods	214
TracingOptions.....	215
TracingOptions properties	215
TracingOptions methods	216
Variable	217
Variable properties	217
Variable methods	217
Variables	218
Variables properties	218
Variables methods	218
View	219
View properties.....	219
Views.....	220
Views properties.....	220
Views methods.....	220
Words.....	221
Words properties.....	221
Words methods	221
Enumerations reference.....	222

This reference section describes the objects and methods in the Illustrator VBScript type library. All of the classes in the type library are presented alphabetically. The chapter concludes with an enumerations reference which lists all of the enumerations in the Illustrator type library.

Each class listing includes the following:

- Properties of the class, including value type, read-only status, and an explanation.
- Methods for the class. Value types needed by the method are shown in bold face. Enumerated values are linked to the [Enumerations reference](#). Required terms are shown in plain face. All items surrounded by brackets [] are optional.
- Script examples.

These example are intended to illustrate concepts, and do not necessarily represent the best or most efficient way to construct a VBScript script. They contain little error checking, and assume that the proper context exists for the scripts to execute in (for instance, that there is a document open or items selected).

Each script contains a single subroutine that can be pasted into any event in a VBScript form if you are using the VBScript development environment. A standard button click event is used for all examples. If you are using a built-in VBScript editor in a VBA application, you can paste the script into a macro routine. In either case, modify the `Sub` statement in the example to work with your situation.

For an overview of how the Illustrator object model is structured, see the *Illustrator Scripting Guide*.

Application

The Adobe Illustrator application object, which contains all other Illustrator objects.

Note: If you have both earlier and later versions of Illustrator installed on the same machine and use the `CreateObject()` or `GetObject()` method to obtain an application reference, use the optional numeric version identifier at the end of the string `"Illustrator.Application"`—when the version identifier is not present, the string refers to the latest installed Illustrator version. To specifically target a version:

- For Illustrator 10, use `"Illustrator.Application.1"`
- For Illustrator CS, use `"Illustrator.Application.2"`
- For Illustrator CS2, use `"Illustrator.Application.3"`
- For Illustrator CS3, use `"Illustrator.Application.4"`

Application properties

Property	Value type	What it is
ActionIsRunning	Boolean	Read-only. If <code>true</code> , an action is still running.
ActiveDocument	Document object	The active (frontmost) document in Illustrator.
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
BrowserAvailable	Boolean	Read-only. If <code>true</code> , a Web browser is available.
ColorSettingsList	Variant	Read-only. The list of color-settings files currently available for use.
DefaultColorSettings	File object	Read-only. The default color-settings file for the current application locale.
Documents	Documents collection object	Read-only. The documents in the application.
FlattenerPresetsList	Variant	Read-only. The list of flattener style names currently available for use.
FreeMemory	Long	Read-only. The amount of unused memory (in bytes) within the Adobe Illustrator partition.
Name	String	Read-only. The application's name (not related to the filename of the application file).
Path	String	Read-only. The file path to the application.
PDFPresetsList	Variant Array of Strings	Read-only. The list of preset PDF-options names available for use.
PPDFileList	Variant Array of Strings	Read-only. The list of PPD files currently available for use.
Preferences	Preference Object	Read-only. The application preferences.
PrinterList	Variant	Read-only. A list of available printers.

Property	Value type	What it is
PrintPresetsList	Variant	Read-only. The list of preset printing-options names currently available for use.
ScriptingVersion	String	Read-only. The version of the Scripting plug-in.
Selection	Variant Array (of objects)	All of the currently selected objects in the active (frontmost) document.
StartupPresetsList	Variant	Read-only. The list of presets available for creating a new document.
TextFonts	TextFonts collection	Read-only. The installed fonts.
TracingPresetList	Variant Array of Strings	Read-only. The list of preset tracing-options names available for use.
UserInteractionLevel	AiUserInteractionLevel	The allowed level of interaction with users in the form of dialogs and message boxes.
Version	String	Read-only. The version of the Adobe Illustrator application.
Visible	Boolean	Read-only. If <code>true</code> , the application is visible.

Application methods

Method	Returns	What it does
ConcatenateMatrix (matrix as Matrix , secondMatrix as Matrix)	Matrix object	Concatenates two matrices.
ConcatenateRotationMatrix (matrix as Matrix , angle as Double)	Matrix object	Concatenates a rotation to a transformation matrix.
ConcatenateScaleMatrix (matrix as Matrix , [, scaleX as Double] [, scaleY as Double])	Matrix object	Concatenates a scale to a transformation matrix.
ConcatenateTranslationMatrix (matrix as Matrix , [, deltaX as Double] [, deltaY as Double])	Matrix object	Concatenates a translation to a transformation matrix.
Copy ()	Nothing	Copies the current selection to the clipboard.
Cut ()	Nothing	Cuts the current selection to the clipboard.
DoJavaScript (JavaScriptCode as String , [, Arguments as Variant Array] [, ExecutionMode as AiJavaScriptExecutionMode])	String	Executes the specified JavaScript code.

Method	Returns	What it does
DoJavaScriptFile (JavaScriptFile as File , [, Arguments as Variant Array] [, ExecutionMode as AiJavaScriptExecutionMode])	String	Executes the specified JavaScript file.
DoScript (Action as String , From as String [, dialogs as Boolean])	Nothing	Plays an action from the Actions palette. Note: Use DoScript only if you intend to run your script from an application external to Illustrator, such as the Windows Script Host. Calling DoScript when running a script from Illustrator's File > Scripts menu may yield unexpected results.
GetIdentityMatrix ()	Matrix object	Returns an identity matrix.
GetPPDFileInfo (Name as String)	PPDFile Info object	Gets detailed file information for specified PPD file.
GetPresetFileOfType (PresetType as AiDocumentPresetType)	File	Returns the full path to the default document profile for the specified preset type.
GetPresetSettings (Preset as String)	Document Preset object	Retrieves the tracing-option settings from the template with a given preset name.
GetRotationMatrix ([Angle as Double])	Matrix object	Returns a transformation matrix containing a single rotation. Note: Requires a value in degrees. For example, 30 rotates the object 30 degrees counterclockwise; -30 rotates the object 30 degrees clockwise.
GetScaleMatrix ([scaleX as Double] [, scaleY as Double])	Matrix object	Returns a transformation matrix containing a single scale. Note: Requires a value in percentage. For example, 60 scales the object to 60 % of its original size; 200 doubles the object's bounds.

Method	Returns	What it does
GetTranslationMatrix ([deltaX as Double] [, deltaY as Double])	Matrix object	Returns a transformation matrix containing a single translation. Note: Requires a value in points. For example, {100,200} moves the object 100 pt. to the right and 200 pt. up; a minus before each number moves the object left and down.
InvertMatrix (matrix as Matrix)	Matrix object	Inverts a matrix.
IsEqualMatrix (matrix as Matrix , secondMatrix as Matrix)	Boolean	Checks whether two matrices are equal.
IsSingularMatrix (matrix as Matrix)	Boolean	Checks whether a matrix is singular and cannot be inverted.
LoadColorSettings (fileSpec as File)	Nothing	Loads color settings from specified file, or, if file is empty, turns color management off.
Open (files as String , [documentColorSpace as AiDocumentColorSpace], [options as Object])	Document object	Opens the file or files specified by the string or array of strings containing file paths. If called to open a pre-Illustrator 9 document that contains both RGB and CMYK colors, supply <code>DocumentColorSpace</code> in order to convert all colors to the specified color space. If not supplied, Illustrator displays a dialog to the user.
Paste ()	Nothing	Pastes the clipboard content into the current document.
Quit ()	Nothing	Quits Illustrator.
Redo ()	Nothing	Redoes the most recently undone transaction.
Redraw ()	Nothing	Forces Illustrator to redraw all its windows.
ShowPresets (fileSpec as String)	Nothing	Gets the presets from the file.

Method	Returns	What it does
TranslatePlaceholderText (text as String)	Nothing	Translates the placeholder text to regular text; a way to enter unicode points in hex values.
Undo ()	Nothing	Undoes the most recent transaction.

► Opening a document

```
' Open a document and get the reference to it
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Open("C:\temp\aiFile.ai")
```

► Accessing the selection

In Illustrator, the application's `Selection` can be accessed as well as modified. The selection will contain `Empty` when there are no selected objects. To deselect all objects in the current document, set the selection to `Empty`, as the following example shows.

```
Set appRef = CreateObject("Illustrator.Application")
appRef.activeDocument.Selection = Empty
```

A reference to a text range is returned when there is an active insertion point in the contents of a `TextFrame`. Similarly, a reference to a range of text is returned when characters are selected in the contents of a `TextFrame`.

► Executing an action

You can run an action from the Action Palette from a script by using the `DoScript` method. When you do this, the control returns to your script before the action has completed. Use the `ActionIsRunning` property to test for when the action has completed before executing any other VBScript methods (see the `Windows.DoAction` example). You can also use the `Sleep` method defined on the `WScript` object to insert a pause to test this property, as in the following example:

```
Set appRef = CreateObject("Illustrator.Application")
appRef.DoScript "Opacity 60 (selection)", "Default Actions"
WHILE (appRef.ActionIsRunning)
    WScript.sleep 1000
WEND
msgbox "Done"
```

► Invoking JavaScript

`DoJavaScript` and `DoJavaScriptFile` can be used to invoke scripts written in JavaScript for Illustrator.

The first argument to `DoJavaScript` is a string that must evaluate to a valid JavaScript expression. For example, the following script displays an alert using the JavaScript `alert` method.

```
appRef.DoJavaScript "alert( 'Number of open documents: ' + _
    documents.length );"
```

This example returns the number of open documents.

```
Set myNumberOfDocuments = appRef.DoJavaScript ("documents.length;")
MsgBox myNumberOfDocuments
```

► Copy and paste between documents

This script uses the application property `ActiveDocument` to copy the current document's selection to the clipboard before pasting it into our new document. This script also demonstrates how to create a new document with a specific color space and dimensions.

```
'Duplicates the selected item in the current document to a new document

Set appRef = CreateObject("Illustrator.Application")

If appRef.Documents.Count > 0 Then
    If Not IsEmpty(appRef.ActiveDocument.selection) Then
        appRef.ActiveDocument.Copy
        Set newDocument = appRef.Documents.Add(aiDocumentCMYKColor, 250, 400)
        newDocument.Paste
    End If
End If
```


Brush

A brush in an Illustrator document. Illustrator brushes can be accessed, but not created or deleted, from a script.

Brush properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Name	String	The Brush name.
Parent	Object	Read-only. The container object.

Brush methods

Method	Returns	What it does
ApplyTo (artItem as PageItem)	Nothing	Applies the brush to a specific art item.

► Applying a Brush

```
'Creates a new group, adds a duplicate of each selected item to the group
'then applies a brush to each item in the group
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
If appRef.Documents.Count > 0 Then
```

```
    If Not IsEmpty(appRef.Selection) Then
```

```
        selectedItems = appRef.Selection
```

```
        startIndex = LBound(selectedItems)
```

```
        endIndex = UBound(selectedItems)
```

```
        Set newGroup = appRef.ActiveDocument.GroupItems.Add
```

```
        For i = startIndex To endIndex
```

```
            Set artItem = selectedItems(i)
```

```
            Set dupItem = artItem.Duplicate(newGroup, aiPlaceAtEnd)
```

```
            appRef.ActiveDocument.Brushes(2).ApplyTo artItem
```

```
        Next
```

```
    End If
```

```
End If
```

Brushes

A collection of brushes in a document.

Brushes properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Long. The number of objects in the collection.
Parent	Object	Read-only. The document that contains this object.

Brushes methods

Method	Returns	What it does
Index (item as Brush)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Brush object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).

► Get all brushes in a document

```
'Counts the number of brushes in the current document  
'and stores in numBrushes  
  
Set appRef = CreateObject("Illustrator.Application")  
  
If appRef.Documents.Count > 0 Then  
    numBrushes = appRef.ActiveDocument.Brushes.Count  
End If
```

CharacterAttributes

Specifies the properties of a character contained in a text frame. A `CharacterStyle` object associates attributes with a text range. See [“CharacterStyle” on page 24](#).

Note: Character attributes do not have default values and are undefined until explicitly set.

CharacterAttributes properties

Property	Value Type	What it is
AkiLeft	Double	The amount of inter-glyph space added to the left side of the glyph in Japanese text (in thousandths of an em).
AkiRight	Double	The amount of inter-glyph spacing added to the right side of the glyph in Japanese text (in thousandths of an em).
Alignment	AiStyleRunAlignmentType	The character alignment type.
AlternateGlyphs	AiAlternateGlyphsForm	Specifies which kind of alternate glyphs are used.
Application	Application object	Read-only. The Illustrator Application object.
AutoLeading	Boolean	If <code>true</code> , automatic leading is used.
BaselineDirection	AiBaselineDirectionType	The Japanese text baseline direction.
BaselinePosition	AiFontBaselineOption	The baseline position of text.
BaselineShift	Double	The amount of shift (in points) of the text baseline.
Capitalization	AiFontCapsOption	Specifies whether the text is normal, all uppercase, all small caps, or a mix of small caps and lowercase.
ConnectionForms	Boolean	If <code>true</code> , OpenType® connection forms are used.
ContextualLigature	Boolean	If <code>true</code> , contextual ligatures are used.
DiscretionaryLigature	Boolean	If <code>true</code> , discretionary ligatures are used.
FigureStyle	AiFigureStyleType	The number style in an OpenType font.
FillColor	Object	The color of the text fill.

Property	Value Type	What it is
Fractions	Boolean	If <code>true</code> , OpenType fractions are used.
HorizontalScale	Double	The horizontal scaling factor (as a percentage).
Italics	Boolean	If <code>true</code> , the Japanese font supports italics.
KerningMethod	AiAutoKernType	The kerning method.
Language	AiLanguageType	The language of the text.
Leading	Double	The amount of space between two lines of text, in points.
Ligature	Boolean	If <code>true</code> , ligatures are used.
NoBreak	Boolean	If <code>true</code> , no break is allowed.
OpenTypePosition	AiFontOpenTypePositionOption	The OpenType baseline position.
Ordinals	Boolean	If <code>true</code> , OpenType ordinals are used.
Ornaments	Boolean	If <code>true</code> , OpenType ornaments are used.
OverprintFill	Boolean	If <code>true</code> , overprint the fill of the text.
OverprintStroke	Boolean	If <code>true</code> , overprinting of the stroke of the text is allowed.
Parent	Object	Read-only. The object's container.
ProportionalMetrics	Boolean	If <code>true</code> , proportional metrics in Japanese OpenType are used.
Rotation	Double	The character rotation angle.
Size	Double	The font size in points.
StrikeThrough	Boolean	If <code>true</code> , characters are struck-through by a line.
StrokeColor	Object	The color of the text stroke.
StrokeWeight	Double	Line width of stroke.
StylisticAlternates	Boolean	If <code>true</code> , OpenType stylistic alternates are used.
Swash	Boolean	If <code>true</code> , OpenType swash characters are used.
TateChuYokoHorizontal	Long	The Tate-Chu-Yoko horizontal adjustment in points.

Property	Value Type	What it is
TateChuYokoVertical	Long	The Tate-Chu-Yoko vertical adjustment in points.
TextFont	TextFont	The font used for the character.
Titling	Boolean	If <code>true</code> , OpenType titling alternates are used.
Tracking	Long	The tracking or range kerning amount in thousandths of an em.
Tsume	Double	The percentage of space reduction around a Japanese character.
Underline	Boolean	If <code>true</code> , characters are underlined.
VerticalScale	Double	Character vertical scaling factor.
WariChuCharactersAfter Break	Long	Specifies how the characters in Wari-Chu text (an inset note in Japanese text) are divided into two or more lines.
WariChuCharactersBefore Break	Long	Specifies how the characters in Wari-Chu text (an inset note in Japanese text) are divided into two or more lines.
WariChuEnabled	Boolean	If <code>true</code> , Wari-Chu is enabled.
WariChuJustification	AiWariChuJustificationType	The Wari-Chu justification.
WariChuLineGap	Long	The Wari-Chu line gap.
WariChuLines	Long	The number of Wari-Chu (multiple text lines fit into a space meant for one) lines.
WariChuScale	Double	The Wari-Chu scale.

► Setting character attributes

```
'Opens a document and adds a simple text frame, then
'uses the CharacterAttributes object to increment
'the horizontal and vertical scale of each character.
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
Set docRef = appRef.Documents.Add()
Set textRef = docRef.TextFrames.Add()
textRef.Contents = "I'd rather be scripting!"
textRef.Top = 500
textRef.Left = 40
```

```
dSize = 100
iCount = textRef.Characters.Count
```

```
i = 1
Dim charRef
Do While (i < (iCount + 1))
    dSize = dSize * 1.1
    textRef.TextRange.Characters(i).CharacterAttributes.HorizontalScale =
dSize
    textRef.TextRange.Characters(i).CharacterAttributes.VerticalScale =
dSize
    i = i + 1
Loop
```

Characters

A collection of `TextRange` objects in which each represents a single character.

Characters properties

Property	Value type	What it is
Application	Application	Read-only. Application that the collection belongs to.
Count	Long	Read-only. Number of elements in the collection.
Parent	Object	Read-only. The object's container.

Characters methods

Method	Returns	What it does
Add (contents as String [, relativeObject as TextFrame] [, insertionLocation as AiElementPlacement])	TextRange	Adds a new character with specified text contents at the specified location in the current document. If location is not specified, adds the new character to the containing text frame after the current text selection or insertion point.
AddBefore (contents as String)	TextRange	Adds a character before the current text selection or insertion point.
Index (itemPtr as TextRange)	Long	Returns the index position of the object within the collection.
RemoveAll ()	Nothing	Deletes every element in the collection.

► Counting characters

```
'Counts the number of characters in the current document
'and stores the result in numChars
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
If appRef.Documents.Count > 0 Then
    numChars = 0
    For Each textArt In appRef.ActiveDocument.TextFrames
        Set textArtRange = textArt.TextRange
        numChars = numChars + textArtRange.length
    Next
End If
```

CharacterStyle

A named style that specifies character attributes. For an example, see [CharacterStyles](#).

CharacterStyle properties

Property	Value type	What it is
Application	Application	Read-only. Application to which the collection belongs.
CharacterAttributes	CharacterAttributes	Read-only. The character properties for the style.
Name	String	The character style's name.
Parent	Object	Read-only. The object's container.

CharacterStyle methods

Method	Returns	What it does
ApplyTo (textItem as Object [, clearingOverrides as Boolean])	Nothing	Applies the character style to the text object or objects.
Delete ()	Nothing	Deletes the object.

CharacterStyles

A collection of character styles.

CharacterStyles properties

Property	Value type	What it is
Application	Application	Read-only. Application to which the collection belongs.
Count	Long	Read-only. Number of elements in the collection.
Parent	Object	Read-only. The object's container.

CharacterStyles methods

Method	Returns	What it does
Add (Name as String)	CharacterStyle	Creates a named character style.
Index (itemPtr as CharacterStyle)	Long	Returns the index position of the object within the collection.
Item (itemKey)	CharacterStyle	Returns an object reference to the object identified by <i>itemKey</i> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Create and apply character styles

```
'Opens a document, adds 3 text frames,
'creates a new character style and applies it to
'each of the text frames
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
'Create a new document add a 3 simple text items
Set docRef = appRef.Documents.Add()
Set textRef1 = docRef.TextFrames.Add()
textRef1.Contents = "Scripting is fun!"
textRef1.Top = 700
textRef1.Left = 50
```

```
Set textRef2 = docRef.TextFrames.Add()
textRef2.Contents = "Scripting is easy!"
textRef2.Top = 625
textRef2.Left = 100
```

```
Set textRef3 = docRef.TextFrames.Add()
textRef3.Contents = "Everyone should script!"
textRef3.Top = 550
textRef3.Left = 150
appRef.Redraw
```

```
'Create a new character style
Set charStyle = docRef.CharacterStyles.Add("BigRed")

'Create a red color
Set colorRed = CreateObject("Illustrator.RGBColor")
colorRed.Red = 255

'Set character attributes of the new style
With charStyle.CharacterAttributes
    .Size = 40
    .Tracking = -50
    .Capitalization = 0 'aiNormalCaps
    .FillColor = colorRed
End With

'Apply style to each textFrame in the document
charStyle.ApplyTo textRef1.TextRange
charStyle.ApplyTo textRef2.TextRange
charStyle.ApplyTo textRef3.TextRange
```

CMYKColor

A CMYK color specification, used to apply a CMYK color to a layer or art item.

If the color space of a document is RGB and you specify the color value for a page item in that document using CMYK, Illustrator translates the CMYK color specification into an RGB color specification. The same thing happens if the document's color space is CMYK and you specify colors using RGB. Since this translation can lose information, you should specify colors using the class that matches the document's actual color space.

CMYKColor properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Black	Double	The black color value. Range: 0.0 to 100.0
Cyan	Double	The cyan color value. Range: 0.0 to 100.0
Magenta	Double	The magenta color value. Range: 0.0 to 100.0
Yellow	Double	The yellow color value. Range: 0.0 to 100.0

► Setting CMYK colors

```
'Creates a new CMYK color and applies it to the first path item
```

```
Set appRef = CreateObject("Illustrator.Application")  
Set newCMYKColor = CreateObject("Illustrator.CMYKColor")
```

```
'Get a reference to the frontmost path in the document  
Set frontPath = appRef.ActiveDocument.PathItems(1)
```

```
'Set color values for the CMYK objects
```

```
newCMYKColor.Black = 0  
newCMYKColor.Cyan = 30.4  
newCMYKColor.Magenta = 32  
newCMYKColor.Yellow = 0
```

```
frontPath.Filled = True  
frontPath.fillColor = newCMYKColor
```

CompoundPathItem

Compound paths are objects composed of multiple intersecting paths, resulting in transparent interior spaces where the original paths overlapped. The `PathItems` property provides access to the paths that make up the compound path.

All paths in a compound path share property values. Therefore, if you set the value of a property of any one of the paths in the compound path, all other paths' matching property will be updated to the new value.

Paths contained within a compound path or group in a document are returned as individual paths when a script asks for the paths contained in the document. However, paths contained in a compound path or group are not returned when a script asks for the paths in a layer which contains the compound path or group.

CompoundPathItem properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
ArtworkKnockout	AiKnockoutState	Is this object used to create a knockout? If so, what kind of knockout?
BlendingMode	AiBlendModes	The mode used when compositing an object.
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Editable	Boolean	Read-only. If <code>true</code> , the object is editable.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object excluding stroke width.
Height	Double	The height of the object, excluding stroke width, calculated from the <code>GeometricBounds</code> .
Hidden	Boolean	If <code>true</code> , this object is hidden.
IsIsolated	Boolean	If <code>true</code> , this object is isolated.
Layer	Layer object	Read-only. The layer to which this object belongs.
Left	Double	The position of the left side of the item (in points, measured from the left side of the page).
Locked	Boolean	If <code>true</code> , the object is locked.
Name	String	The name of this object.
Note	String	The note assigned to this object.
Opacity	Double	The opacity of the object. Range: 0.0 to 100.0
Parent	Document object	Read-only. The document that contains this object.
PathItems	PathItems collection object	Read-only. The path art items in this compound path.

Property	Value type	What it is
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item, in the format [x, y]. Does not include stroke weight.
Selected	Boolean	If <code>true</code> , the object is selected.
Sliced	Boolean	If <code>true</code> , the <code>CompoundPathItem</code> is sliced. Default: <code>false</code>
Tags	Tags collection object	Read-only. The tags contained in this object.
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).
URL	String	The value of the Adobe URL tag assigned to this object.
VisibilityVariable	Variable	The <code>VisibilityVariable</code> bound to this object.
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the object including stroke width.
Width	Double	The width of the object, excluding stroke width, calculated from the <code>GeometricBounds</code> .
WrapInside	Boolean	If <code>true</code> , the text frame object are wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If <code>true</code> , wrap text frame objects around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this art item within the stacking order of the group or layer (<code>Parent</code>) that contains the art item.

CompoundPathItem methods

Method	Returns	What it does
Copy ()	Nothing	Copies the compound path to the clipboard. The associated document must be the frontmost document.
Cut ()	Nothing	Cuts the compound path onto the clipboard. The associated document must be the frontmost document.
Delete ()	Nothing	Deletes the object.
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	Object	Duplicates the compound path, optionally specifying the location and position for the copy.

Method	Returns	What it does
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the object, specifying the new location and position.
Resize (scaleX as Double , scaleY as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the compound path where <code>scaleX</code> is the horizontal scaling factor (as a percentage) and <code>scaleY</code> is the vertical scaling factor; 100.0 = 100%
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the art item relative to the current rotation. The object is rotated counter-clockwise if the <code>Angle</code> value is positive, clockwise if the value is negative.
Transform (transformationMatrix as Matrix , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the compound path by applying the specified transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the compound path relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the compound path's position in the stacking order of the group or layer (<code>Parent</code>) that contains the object.

► Selecting paths in a document

This example demonstrates how to select all of the paths in a document that are not part of a compound path or a group by testing the type of the `Parent` property with a `TypeName` function.

```
'Selects path items in the current document that are not
'compound paths

Set appRef = CreateObject("Illustrator.Application")
counter = 0
If appRef.Documents.Count > 0 Then
    If appRef.ActiveDocument.PathItems.Count > 0 Then
        For Each pathArt In appRef.ActiveDocument.PathItems
            If (Not TypeName(pathArt.Parent) = "CompoundPathItem") Then
                pathArt.Selected = True
                counter = counter + 1
            Else
                pathArt.Selected = False
            End If
        Next
    End If
End If
```

► Creating a compound path

This example demonstrates how to create a new compound path containing three `PathItems`. The example then modifies the stroke of the paths in the compound path. Note that when you modify the properties of a `PathItem` inside a compound path you affect all paths contained in the compound path. The example also shows how to access swatches in a document by name.

```
'Creates a new compound path item, adds 3 new
'path items, then sets the stroke of the compound path

Set appRef = CreateObject("Illustrator.Application")

Set frontDocument = appRef.ActiveDocument
Set activeLayer = frontDocument.activeLayer
Set newCompoundPath = activeLayer.CompoundPathItems.Add

'Create the path items
Set newPath = newCompoundPath.PathItems.Add
newPath.SetEntirePath Array(Array(30, 50), Array(30, 100))

Set newPath = newCompoundPath.PathItems.Add
newPath.SetEntirePath Array(Array(40, 100), Array(100, 100))

Set newPath = newCompoundPath.PathItems.Add
newPath.SetEntirePath Array(Array(100, 110), Array(100, 300))

'Set the stroke of the compound path
newPath.Stroked = True
newPath.StrokeWidth = 3.5
newPath.StrokeColor = frontDocument.Swatches(8).Color
```

CompoundPathItems

A collection of compound paths.

CompoundPathItems properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Count	Long	Read-only. The number of objects in the collection.
Parent	Document object	Read-only. The document that contains this object.

CompoundPathItem methods

Method	Returns	What it does
Add ()	CompoundPathItem object	Creates a new CompoundPathItem object.
Index (item as CompoundPathItem)	Long	Returns the index position of the object within the collection.
Item (itemKey)	CompoundPathItem object	Returns an object reference to the object identified by itemKey (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Counting compound paths

```
'Counts the compound path items in the current document and
'stores in numCPaths

Set appRef = CreateObject("Illustrator.Application")

If appRef.Documents.Count > 0 Then
    numCPaths = appRef.ActiveDocument.Layers(1).CompoundPathItems.Count
End If
```


DataSet

A set of data used for dynamic publishing. A dataset allows you to collect a number of variables and their dynamic data into one object. You must have at least one variable bound to an art item in order to create a DataSet object.

- See the [Variable](#) class, and "Working with Variables and Datasets" in the *Illustrator Scripting Guide* for more information.
- See [DataSets](#) for a sample script that uses a DataSet object.

DataSet properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Name	String	The name of the DataSet.
Parent	Document object	Read-only. The name of the object that is this DataSet object's parent.

DataSet methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.
Display ()	Nothing	Displays the DataSet.
Update ()	Nothing	Updates the DataSet.

DataSets

A collection of DataSet objects.

DataSets properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Count	Long	Read-only. The number of datasets in the collection.
Parent	Document object	Read-only. The name of the object that is this DataSet object's parent.

DataSets methods

Method	Returns	What it does
Add ()	DataSet object	Creates a new DataSet object.
Index (item as DataSets)	Long	Returns the index position of the object within the collection.
Item (itemKey)	DataSet object	Returns an object reference to the object identified by itemKey (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Creating datasets

```
'Creates two variables (visibility and text),
'creates two datasets using different variable
'values then displays each dataset

Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()

'Create a visibility variable
Set itemRef = docRef.PathItems.Rectangle(600, 200, 150, 150)
Set colorRef = CreateObject("Illustrator.RGBColor")
colorRef.Red = 255
itemRef.FillColor = colorRef
Set visibilityVar = docRef.Variables.Add()
visibilityVar.Kind = 2 'aiVisibility
itemRef.VisibilityVariable = visibilityVar

'Create a text variable
Set textRef = docRef.TextFrames.Add
textRef.Contents = "Text Variable, dataset 1"
textRef.Top = 400
textRef.Left = 200
Set textVar = docRef.Variables.Add
```

```
textVar.Kind = 3 'aiTextual
textRef.ContentVariable = textVar

'Create dataset 1
Set ds1 = docRef.DataSets.Add()

'Change data and create dataset 2
itemRef.Hidden = True
textRef.Contents = "Text Variable, dataset 2"
Set ds2 = docRef.DataSets.Add()

'Display datasets
ds1.Display
ds2.Display
```

Document

An Illustrator document. Documents are contained in the `Application` object.

The default document settings—those properties starting with the word “Default”—are global settings that affect the current document. Be sure to modify these default properties only when a document is open. Note that if you set default properties to desired values before creating new objects, you can streamline your scripts, eliminating the need to specify properties such as `FillColor` and `Stroked` that have analogous default properties.

A document’s `DocumentColorSpace`, `Height`, and `Width` can be set only when the document is created. Once a document is created, these properties cannot be changed.

The frontmost document can be referred to as either `AppRef.ActiveDocument` or `AppRef.Documents(1)`.

Document properties

Property	Value type	What it is
ActiveDataSet	<code>DataSet</code> object	The active <code>DataSet</code> object in the document.
ActiveLayer	<code>Layer</code> object	The active <code>Layer</code> in the document.
ActiveView	<code>View</code> object	Read-only. The document’s current <code>View</code> .
Application	<code>Application</code> object	Read-only. The Illustrator <code>Application</code> object.
Brushes	<code>Brushes</code> collection object	Read-only. The <code>Brushes</code> contained in the document.
CharacterStyles	<code>CharacterStyles</code>	Read-only. The list of character styles in this document.
CompoundPathItems	<code>CompoundPathItems</code> collection object	Read-only. The <code>CompoundPathItems</code> contained in the document.
CropBox	Variant Array of 4 <code>Doubles</code>	The boundary of the document’s cropping box for output. A document does not have a default <code>CropBox</code> . In order to read this property you have to set the <code>CropBox</code> first.
CropStyle	AiCropOptions	The style of the document’s cropping box.
DataSets	<code>DataSet</code> object	Read-only. A <code>DataSets</code> collection in the document.
DefaultFillColor	<code>Color</code> object	The <code>Color</code> to fill new paths if default filled is <code>true</code> .
DefaultFilled	<code>Boolean</code>	If <code>true</code> , a new path are filled.
DefaultFillOverprint	<code>Boolean</code>	If <code>true</code> , the art beneath a filled object are overprinted by default.
DefaultStrokeCap	AiStrokeCap	Default type of line capping for paths created.

Property	Value type	What it is
DefaultStrokeColor	Color object	The stroke color for new paths if default stroked is <code>true</code> .
DefaultStroked	Boolean	If <code>true</code> , the new path are stroked.
DefaultStrokeDashes	Variant Array of Doubles	The default lengths for dashes and gaps in dashed lines, starting with the first dash length, followed by the first gap length, and so on.
DefaultStrokeDashOffset	Double	The default distance into the dash pattern at which the pattern are started for new paths.
DefaultStrokeJoin	AiStrokeJoin	Default type of joints in new paths.
DefaultStrokeMiterLimit	Double	When a default stroke join is set to <code>mitered</code> , this property specifies when the join will be converted to beveled (squared-off) by default. The default miter limit of 4 means that when the length of the point reaches four times the stroke weight, the join switches from a miter join to a bevel join. Range: 1 to 500; a value of 1 specifies a bevel join.
DefaultStrokeOverprint	Boolean	If <code>true</code> , the art items beneath a stroked object are overprinted by default.
DefaultStrokeWidth	Double	The default width of stroke for new paths.
DocumentColorSpace	AiDocumentColorSpace	Read-only. The color specification system to use for this document's color space.
FullName	String	Read-only. The file associated with the document, which includes the complete path to the file.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the illustration excluding the stroke width of any objects in the document.
Gradients	Gradients collection object	Read-only. The <code>Gradients</code> collection contained in the document.
GraphicStyles	GraphicStyles collection object	Read-only. The <code>GraphicStyles</code> contained in the document.
GraphItems	collection object	Read-only. The <code>GraphItems</code> collection contained in the document.
GroupItems	GroupItems collection object	Read-only. The <code>GroupItems</code> collection contained in the document.
Height	Double	Read-only. The height of the document, calculated from the <code>GeometricBounds</code> .
InkList	Variant Array	Read-only. The list of inks in this document.

Property	Value type	What it is
KinsokuSet	Object	Read-only. The Kinsoku set of characters that cannot begin or end a line of Japanese text.
Layers	Layers collection object	Read-only. The layers contained in the document.
LegacyTextItems	LegacyTextItems	Read-only. The legacy text items in the document.
MeshItems	MeshItems collection object	Read-only. The mesh art items contained in the document.
MojikumiSet	Object	Read-only. A list of predefined Mojikumi sets that specify the spacing for the layout and composition of Japanese text.
Name	String	Read-only. The document's name (not the complete file path to the document).
OutputResolution	Double	Read-only. The current output resolution for the document in dots per inch (dpi).
PageItems	PageItems collection object	Read-only. The PageItems (contains all art item classes) contained in the document.
PageOrigin	Variant Array of 2 Doubles	The zero-point of the page in the document without margins, relative to the overall height and width.
ParagraphStyles	ParagraphStyles	Read-only. The list of paragraph styles in this document.
Parent	Application object	Read-only. The application that contains this document.
Path	String	Read-only. The file associated with the document, which includes the complete path to the file.
PathItems	PathItems collection object	Read-only. The PathItems contained in this document.
Patterns	Patterns collection object	Read-only. The patterns contained in this document.
PlacedItems	PlacedItems collection object	Read-only. The PlacedItems contained in this document.
PluginItems	PluginItems collection object	Read-only. The PluginItems contained in this document.
PrintTiles	Boolean	Read-only. If <code>true</code> , this document is tiled when printed.
RasterItems	RasterItems collection object	Read-only. The raster items contained in this document.

Property	Value type	What it is
RulerOrigin	Variant Array of 2 Doubles	The zero-point of the rulers in the document relative to the bottom left corner.
RulerUnits	AiRulerUnits	Read-only. The default measurement units for the rulers in the document.
Saved	Boolean	If <code>true</code> , the document has not been changed since the last time it was saved.
Selection	Variant Array of objects	The array of references to the objects in this document's current selection.
ShowPlacedImages	Boolean	Read-only. If <code>true</code> , placed images are displayed in the document.
SplitLongPaths	Boolean	Read-only. If <code>true</code> , long paths are split when printing.
Spots	Spots collection object	Read-only. The <code>SpotColors</code> contained in this document.
Stationery	Boolean	Read-only. If <code>true</code> , the file is a stationery file.
Stories	Stories	Read-only. The story items in this document.
Swatches	Swatches collection object	Read-only. The <code>Swatches</code> contained in this document.
SymbolItems	SymbolItems collection object	Read-only. The <code>SymbolItems</code> collection contained in this document.
Symbols	Symbols collection object	Read-only. The <code>Symbols</code> collection contained in this document.
Tags	Tags collection object	Read-only. The <code>Tags</code> contained in this document.
TextFrames	TextFrames collection object	Read-only. The <code>TextFrames</code> contained in this document.
TileFullPages	Boolean	Read-only. If <code>true</code> , full pages are tiled when printing this document.
UseDefaultScreen	Boolean	Read-only. If <code>true</code> , the printer's default screen is used when printing this document.
Variables	Variables collection object	Read-only. The <code>Variables</code> collection contained in this document.
VariablesLocked	Boolean	If <code>true</code> , the <code>Variables</code> in this document are locked.
Views	Views collection object	Read-only. The <code>Views</code> contained in this document.

Property	Value type	What it is
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the document, including stroke width of any objects in the illustration.
Width	Double	Read-only. The width of this document, calculated from the <code>GeometricBounds</code> .
XMPString	String	The XMP metadata packet associated with this document.

Document methods

Method	Returns	What it does
Activate ()	Nothing	Bring the first window associated with the document to the front
Close ([saving as AiSaveOptions])	Nothing	Closes the document
Copy ()	Nothing	Copies the current selection in the document to the clipboard; the associated document must be the frontmost document
Cut ()	Nothing	Cuts the current selection in the document to the clipboard; the associated document must be the frontmost document
ExportPDFPreset (File as a String)	Nothing	Save all PDF presets to a file
Export (exportFile as String , exportFormat as AiExportType , [options as Object])	Nothing	Exports the document to the specified file using one of the export file formats. Options for various formats are collected in <code>ExportOptionsFormat</code> objects. The appropriate file extension is automatically appended to the file name, except for Photoshop® documents. For these, you must include the file extension (<code>.psd</code>) in the file specification.
ExportPDFPreset ()	String	Exports the current PDF preset values to the file
ExportPrintPreset ()	String	Exports the current print preset values to the file
ExportVariables (Filename as String)	Nothing	Exports <code>Variables</code> from this document to a specified file
ImageCapture (imageFile as File, clipBounds as Rect, options as ImageCaptureOptions)	Nothing	Captures the artwork content within the clipping boundaries in this document as a raster image, and writes the image data to a specified file.

Method	Returns	What it does
ImportCharacterStyles (FileSpec as String)	Nothing	Loads the character styles from the Illustrator file.
ImportParagraphStyles (FileSpec as String)	Nothing	Loads the paragraph styles from the Illustrator file.
ImportPDFPreset (FileSpec as String [, ReplacingPreset as Boolean])	Nothing	Loads all PDF presets from a file.
ImportPrintPreset (printPreset as String FileSpec as String)	Nothing	Loads the named print preset from the file.
ImportVariables (filename as String)	Nothing	Imports <code>Variables</code> from the specified file into this document.
Paste ()	Nothing	Pastes the contents of the clipboard into the current layer of the document; if the document is the frontmost then all pasted objects remain selected after the paste.
PrintOut (options as PrintOptions)	Nothing	Prints the document.
Save ()	Nothing	Saves the document in its current location.
SaveAs ([saveIn as String] [, options As: EPSSaveOptions / IllustratorSaveOptions / PDFSaveOptions])	Nothing	Saves the document in the specified file as an Illustrator, EPS, or PDF file.

► Closing documents

If you close the document, you should set your document reference to `Nothing` to prevent your script accidentally trying to access closed documents.

```
'Closes the frontmost document without saving
```

```
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents(1)
```

```
docRef.Close 2' aiDoNotSaveChanges
Set docRef = Nothing
```

➤ Creating a document with defaults

```
'Sets the default stroke and fill in the current document to false
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
If (appRef.Documents.Count = 0) Then
```

```
    Set frontDocument = appRef.Documents.Add
```

```
Else
```

```
    Set frontDocument = appRef.Documents(1)
```

```
End If
```

```
frontDocument.DefaultFilled = False
```

```
frontDocument.DefaultStroked = False
```

Documents

A collection of documents.

Documents properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Count	Long	Read-only. The number of objects in the collection.
Parent	Object	Read-only. The parent of this object.

Documents methods

Method	Returns	What it does
Add ([documentColorSpace As AiDocumentColorSpace] [, width as Double] [, height as Double])	Document object	Creates a new document using optional parameters and returns a reference to the new document.
AddDocument ([startupPreset as String] [, presetSettings as DocumentPreset])	Document object	Creates a new document using optional parameters and returns a reference to the new document.
Index (item as Document)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Document object	Returns an object reference to the object identified by itemKey (name or index).

► Creating a document with a color space

'Creates a new RGB document

```
Set appRef = CreateObject("Illustrator.Application")
appRef.Documents.Add 1 'aiDocumentRGBColor
```

DocumentPreset

A preset document template to use when creating a new document with [AddDocument](#).

DocumentPreset properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
ColorMode	AiDocumentColorSpace	The color space for the new document.
Height	Double	The height in points. Default is 792.0
PreviewMode	AiDocumentPreviewMode	The preview mode for the new document.
RasterResolution	AiDocumentRasterResolution	The raster resolution for the new document.
Title	String	The document title.
TransparencyGrid	AiDocumentTransparencyGrid	The transparency grid color for the new document.
Units	AiRulerUnits	The ruler units for the new document.
Width	Double	The width in points. Default is 612.0

EPSSaveOptions

Options that may be supplied when saving a document as an Illustrator EPS file. `EPSSaveOptions` can only be supplied in conjunction with the document `SaveAs` method. All properties are optional.

EPSSaveOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
CMYKPostScript	Boolean	If <code>true</code> , CMYK PostScript® is used.
Compatibility	AiCompatibility	Illustrator version compatibility for the EPS file format. Default: 13 ' aiIllustrator13
CompatibleGradientPrinting	Boolean	If <code>true</code> , a raster item of the gradient or gradient mesh are created so the object can be printed on PostScript level 2 printers. Default: <code>false</code>
EmbedAllFonts	Boolean	If <code>true</code> , fonts are included in the EPS file.
EmbedLinkedFiles	Boolean	If <code>true</code> , linked image files are included in the saved document.
FlattenOutput	AiOutputFlattening	How should transparency be flattened for file formats older than Illustrator 9 or greater.
IncludeDocumentThumbnails	Boolean	If <code>true</code> , the thumbnail image of the EPS artwork is included.
Overprint	AiPDFOverprint	The style of overprinting. Default: 1 ' aiPreservePDFOverprint
PostScript	AiEPSPostScriptLevelEnum	Specifies the PostScript level to use when saving the file.
Preview	AiEPSPreviewX	Format for the EPS preview image.

► Saving to EPS

```
'Saves the current document to dest in EPS format
'dest contains the full path and file name to save to

Sub exportFileAsEPS (dest)
    Set appRef = CreateObject("Illustrator.Application")
    Set newSaveOptions = CreateObject("Illustrator.EPSSaveOptions")

    If appRef.Documents.Count > 0 Then
        newSaveOptions.CMYKPostScript = True
        newSaveOptions.EmbedAllFonts = True
        Set frontDocument = appRef.ActiveDocument
        frontDocument.SaveAs dest, newSaveOptions
    End If
End Sub
```

ExportOptionsAutoCAD

Options for exporting a document as an AutoCAD file, used with the [Export](#) method. All properties are optional.

When you export a document, the default file extension (DWG) is appended automatically. You should not include a file extension in the file specification. To override the default AutoCAD export format, use the [ExportFileFormat](#) property.

ExportOptionsAutoCAD properties

Property	Value type	What it is
AlterPathsForAppearance	Boolean	If <code>true</code> , paths are altered (if needed) to maintain appearance. Default: <code>false</code>
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Colors	AiAutoCADColors	The number of colors to export into the AutoCAD file.
ConvertTextToOutlines	Boolean	If <code>true</code> , text is converted to outlines. Default: <code>false</code>
ExportFileFormat	AiAutoCADExportFileFormat	The file format to which to export. Default: <code>1 'aiDWG</code>
ExportOption	AiAutoCADExportOption	Whether to preserve appearance or editability during export. Default: <code>1 ' aiMaximumEditability</code>
ExportSelectedArtOnly	Boolean	If <code>true</code> , only selected artwork is exported. Default: <code>false</code>
RasterFormat	AiAutoCADRasterFormat	The format in which to export raster art.
ScaleLineweights	Boolean	If <code>true</code> , line weights are scaled by the same scaling factor as the rest of the drawing. Default: <code>false</code>
Unit	AiAutoCADUnit	Measurement units from which to map.
UnitScaleRatio	Double	The ratio by which to scale output.
Version	AiAutoCADCompatibility	The release of AutoCAD to export to.

ExportOptionsFlash

Specifies options that you can supply when exporting a document as Macromedia® Flash™ (SWF) with the [Document.Export](#) method.

When you export a document, the appropriate file extension is appended automatically. You should not include a file extension in the file specification.

ExportOptionsFlash properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
ArtClipping	AiArtClippingOption	How the art is clipped during the output. Default: 1 ' aiOutputArtBounds
BackgroundColor	RGBColor object	The background color of the exported Flash frames.
BackgroundLayers	Variant Array of Layer objects	A list of layers to be included as the static background of the exported Flash frames.
BlendAnimation	AiBlendAnimationType	How the blend art objects are animated when exported to Flash frames. Default: 0 ' aiNoBlendAnimation
Compressed	Boolean	If <code>true</code> , the exported file is compressed when exported. Default: <code>false</code>
ConvertTextToOutlines	Boolean	If <code>true</code> , all text is converted to outlines. Default: <code>false</code>
CurveQuality	Long	The amount of curve information that is preserved. Range: 0 to 10, Default: 7
ExportAllSymbols	Boolean	If <code>true</code> , all symbols defined in the palette are exported. Default: <code>false</code>
ExportStyle	AiFlashExportStyle	The style in which the exported data are created in Flash. Default: 1 ' aiAsFlashFile
ExportVersion	AiFlashExportVersion	The version of the exported SWF file. Default: 9 ' aiFlashVersion9
FrameRate	Double	The display rate in frames per second. Range: 0.01 to 12.0; Default: 12.0
IgnoreTextKerning	Boolean	If <code>true</code> , kerning information is ignored in text objects. Default: <code>false</code>
ImageFormat	AiFlashImageFormat	How the image in the exported Flash file is compressed. Default: 1 ' aiLossless
IncludeMetadata	Boolean	If <code>true</code> , includes minimal XMP metadata in the SWF file. Default: <code>false</code>

Property	Value type	What it is
JPEGMethod	AiFlashJPEGMethod	The JPEG method to use. Default: 1 ' aiJPEGStandard
JPEGQuality	Long	Level of compression to use. Range: 0 to 10. Default: 3
LayerOrder	AiLayerOrderType	The order in which layers are exported to Flash frames. Default: 1 ' aiBottomUp
Looping	Boolean	If <code>true</code> , the Flash file are set to loop when run. Default: <code>false</code>
PlaybackAccess	AiFlashPlaybackSecurity	The access level for the exported SWF file. Default: 1 ' aiPlaybackLocal
PreserveAppearance	Boolean	If <code>true</code> , preserve appearance. If <code>false</code> , preserve editability. Default: <code>false</code>
ReadOnly	Boolean	If <code>true</code> , the file is exported as read-only. Default: <code>false</code>
Replacing	AiSaveOptions	If a file with the same name already exists, should it be replaced. Default: 1 ' aiPromptToSaveChanges
Resolution	Double	The resolution in pixels per inch. Range: 72 to 2400. Default: 72

ExportOptionsGIF

Options that may be supplied when exporting a document as a GIF file. See the [Document Export](#) method for additional details.

When you export a document, the appropriate file extension is appended automatically. You should not include a file extension in the file specification.

ExportOptionsGIF properties

Property	Value type	What it is
AntiAliasing	Boolean	If <code>true</code> , the exported image is anti-aliased. Default: <code>true</code>
Application	Application object	Read-only. The Illustrator Application object.
ArtBoardClipping	Boolean	If <code>true</code> , the exported image is clipped to the art board. Default: <code>false</code>
ColorCount	Long	The number of colors in the exported image's color table. Range: 2 to 256. Default: 128
ColorDither	AiColorDitherMethod	The method used to dither colors in the exported image. Default: <code>&H4466736E ' aiDiffusion</code>
ColorReduction	AiColorReductionMethod	The method used to reduce the number of colors in the exported image. Default: <code>&H536C7476 ' aiSelective</code>
DitherPercent	Long	The amount that the colors of the exported image are dithered, where 100.0 is 100%
HorizontalScale	Double	The horizontal scaling factor applied to the exported image, where 100.0 is 100%. Default: 100.0
InfoLossPercent	Long	The level of information loss allowed during compression, where 100.0 is 100%. Default: 0
Interlaced	Boolean	If <code>true</code> , the exported image is interlaced. Default: <code>false</code>
Matte	Boolean	If <code>true</code> , the art board is matted with a color. Default: <code>true</code>
MatteColor	RGBColor object	The color to use when matting the art board. Default: <code>white</code>
SaveAsHTML	Boolean	If <code>true</code> , the exported image is saved with an accompanying HTML file. Default: <code>false</code>
Transparency	Boolean	If <code>true</code> , the exported image uses transparency. Default: <code>true</code>

Property	Value type	What it is
VerticalScale	Double	The vertical scaling factor applied to the exported image, where 100.0 is 100%. Default: 100.0
WebSnap	Long	How much should the color table be changed to match the web palette, where 100 is maximum. Default: 0

► Exporting as GIF

```
'Exports the current document as GIF to dest
'dest contains the full path and file name to save to

Sub exportFileAsGIF (dest)
    Set appRef = CreateObject("Illustrator.Application")
    Set gifExportOptions = CreateObject("Illustrator.ExportOptionsGIF")

    If appRef.Documents.Count > 0 Then
        gifExportOptions.AntiAliasing = False
        gifExportOptions.ColorCount = 64
        gifExportOptions.ColorDither = 1147564910' aiDiffusion
        Set docRef = appRef.ActiveDocument
        Call docRef.Export (dest, 6, gifEportOptions)' 6 = aiGIF
    End If
End Sub
```

ExportOptionsJPEG

Options that may be supplied when exporting a document as a JPEG file. See the [Document Export](#) method for additional details.

When you export a document, the appropriate file extension is appended automatically. You should not include a file extension in the file specification.

ExportOptionsJPEG

Property	Value type	What it is
AntiAliasing	Boolean	If <code>true</code> , the exported image is anti-aliased. Default: <code>true</code>
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
ArtBoardClipping	Boolean	If <code>true</code> , the exported image is clipped to the art board. Default: <code>false</code>
BlurAmount	Double	The amount of blur to apply to the exported image. Range: 0.0 to 2.0. Default: <code>0.0</code>
HorizontalScale	Double	The horizontal scaling factor applied to the exported image, where 100.0 is 100%. Default: <code>100.0</code>
Matte	Boolean	If <code>true</code> , the art board is matted with a color. Default: <code>true</code>
MatteColor	RGBColor object	The color to use when matting the art board. Default: <code>white</code>
Optimization	Boolean	If <code>true</code> , the exported image is optimized for web viewing. Default: <code>true</code>
QualitySetting	Long	The quality of the exported image. Range: 0 to 100. Default: <code>30</code>
SaveAsHTML	Boolean	If <code>true</code> , the exported image is saved with an accompanying HTML file. Default: <code>false</code>
VerticalScale	Double	The vertical scaling factor applied to the exported image, where 100.0 is 100%. Default: <code>100.0</code>

► Exporting to JPEG

```
'Exports the current document as JPEG to dest
'dest contains the full path and file name to save to

Sub exportFileAsJPEG (dest)
    Set appRef = CreateObject("Illustrator.Application")
    Set jpegExportOptions = CreateObject("Illustrator.ExportOptionsJPEG")

    If appRef.Documents.Count > 0 Then
        jpegExportOptions.AntiAliasing = False
        jpegExportOptions.QualitySetting = 70
        Set docRef = appRef.ActiveDocument
        Call docRef.Export (dest, 1, jpegExportOptions) ' 1 = aiJPEG
    End If
End Sub
```

ExportOptionsPhotoshop

Options that may be supplied when exporting a document as an Adobe Photoshop® file. See the [Document Export](#) method on page 40 for additional details.

When you export a document, the appropriate file extension is appended automatically. You should not include a file extension in the file specification.

ExportOptionsPhotoshop properties

Property	Value type	What it is
AntiAliasing	Boolean	If <code>true</code> , the exported image is anti-aliased. Default: <code>true</code>
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
EditableText	Boolean	If <code>true</code> , text objects are exported as editable text layers. Default: <code>true</code>
EmbedICCProfile	Boolean	If <code>true</code> , an ICC profile are embedded in the exported file. Default: <code>false</code>
ImageColorSpace	Ai ImageColorSpace	The color space of the exported file. Default: <code>aiImageRGB ' 2</code>
MaximumEditability	Boolean	If <code>true</code> , preserve as much of the original document's structure as possible when exporting. Default: <code>true</code>
Resolution	Double	The resolution of the exported file in dots per inch. Range: 72 to 2400. Default: 150
Warnings	Boolean	If <code>true</code> , a warning dialog are displayed because of conflicts in the export settings. Default: <code>true</code>
WriteLayers	Boolean	If <code>true</code> , the document layers are preserved in the exported file. Default: <code>true</code>

► Exporting to Photoshop

This example exports the current document as a Photoshop 5 file with layers.

```
'Exports the current document as PSD (Photoshop) to dest
'dest contains the full path and file name to save to

Sub exportFileAsPSD (dest)
    Set appRef = CreateObject("Illustrator.Application")
    Set psdExportOptions =
    CreateObject("Illustrator.ExportOptionsPhotoshop")

    If appRef.Documents.Count > 0 Then
        psdExportOptions.Resolution = 150
        Set docRef = appRef.ActiveDocument
        Call docRef.Export (dest, 2, psdExportOptions) ' 2 = aiPhotoshop
    End If
End Sub
```

ExportOptionsPNG8

Options that may be supplied when exporting a document as an 8-bit PNG file. See the [Document Export](#) method for additional details.

When you export a document, the appropriate file extension is appended automatically. You should not include a file extension in the file specification.

ExportOptionsPNG8 properties

Property	Value type	What it is
AntiAliasing	Boolean	If <code>true</code> , the exported image is anti-aliased. Default: <code>true</code>
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
ArtBoardClipping	Boolean	If <code>true</code> , the exported image is clipped to the art board. Default: <code>false</code>
ColorCount	Long	The number of colors in the exported image's color table. Range: 2 to 256. Default: 128
ColorDither	AiColorDitherMethod	The method used to dither colors in the exported image. Default: <code>&H4466736E ' aiDiffusion</code>
ColorReduction	AiColorReductionMethod	The method used to reduce the number of colors in the exported image. Default: <code>&H536C7476 ' aiSelective</code>
DitherPercent	Long	Specifies how much the colors of the exported image are dithered, where 100.0 is 100%
HorizontalScale	Double	The horizontal scaling factor applied to the exported image, where 100.0 is 100%. Default: 100.0
Interlaced	Boolean	If <code>true</code> , the exported image are interlaced. Default: <code>false</code>
Matte	Boolean	If <code>true</code> , the art board is matted with a color. Default: <code>true</code>
MatteColor	RGBColor object	The color to use when matting the art board. Default: <code>white</code>
SaveAsHTML	Boolean	If <code>true</code> , the exported image is saved with an accompanying HTML file. Default: <code>false</code>
Transparency	Boolean	If <code>true</code> , the exported image uses transparency. Default: <code>true</code>
VerticalScale	Double	The vertical scaling factor applied to the exported image, where 100.0 is 100%. Default: 100.0
WebSnap	Long	Specifies how much the color table are changed to match the Web palette, where 100 is maximum. Default: 0

► Exporting to PNG8

This example exports the current document as a PNG8 file.

```
'Exports the current document as PNG8 to dest
'dest contains the full path and file name to save to

Sub exportFileAsPNG (dest)
    Set appRef = CreateObject("Illustrator.Application")
    Set png8ExportOptions = CreateObject("Illustrator.ExportOptionsPNG8")

    If appRef.Documents.Count > 0 Then
        png8ExportOptions.AntiAliasing = False
        png8ExportOptions.Interlaced = True
        Set docRef = appRef.ActiveDocument
        Call docRef.Export (dest, 4, png8ExportOptions) ' 4 = aiPNG8
    End If
End Sub
```


ExportOptionsPNG24

Options that may be supplied when exporting a document as a 24-bit PNG file. See the [Document Export](#) method for additional details.

When you export a document, the appropriate file extension is appended automatically. You should not include a file extension in the file specification.

ExportOptionsPNG24 properties

Property	Value type	What it is
AntiAliasing	Boolean	If <code>true</code> , the exported image is anti-aliased. Default: <code>true</code>
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
ArtBoardClipping	Boolean	If <code>true</code> , the exported image is clipped to the art board. Default: <code>false</code>
HorizontalScale	Double	The horizontal scaling factor applied to the exported image, where 100.0 is 100%. Default: 100.0
Matte	Boolean	If <code>true</code> , the art board is matted with a color. Default: <code>true</code>
MatteColor	RGBColor object	The color to use when matting the art board. Default: <code>white</code>
SaveAsHTML	Boolean	If <code>true</code> , the exported image is saved with an accompanying HTML file. Default: <code>false</code>
Transparency	Boolean	If <code>true</code> , the exported image uses transparency. Default: <code>true</code>
VerticalScale	Double	The vertical scaling factor applied to the exported image, where 100.0 is 100%. Default: 100.0

► Exporting to PNG24

This example exports the current document as a PNG24 file with specific options.

```
'Exports current document as PNG24 to dest
'dest contains the full path and file name to save to

Sub exportFileAsPNG (dest)
    Set appRef = CreateObject("Illustrator.Application")
    Set png24ExportOptions = CreateObject("Illustrator.ExportOptionsPNG24")

    If appRef.Documents.Count > 0 Then
        png24ExportOptions.AntiAliasing = False
        png24ExportOptions.Transparency = False
        Set docRef = appRef.ActiveDocument
        Call docRef.Export (dest, 5, png24ExportOptions) ' 5 = aiPNG24
    End If
End Sub
```

ExportOptionsSVG

Options that may be supplied when exporting a document as a SVG file. See the [Document Export](#) method for additional details.

When you export a document, the appropriate file extension is appended automatically. You should not include a file extension in the file specification.

ExportOptionsSVG properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Compressed	Boolean	If <code>true</code> , the exported file is compressed. Default: <code>false</code>
CoordinatePrecision	Long	The decimal precision for element coordinate values. Range: 1 to 7. Default: 3
CSSProperties	AiSVGCSSPropertyLocation	Specifies how the CSS properties of the document are included in the exported file
DocumentEncoding	AiSVGDocumentEncoding	Specifies how the text in the document is encoded. Default: 1 ' aiASCII
DTD	AiSVGDTDVersion	The SVG version to which the file should conform. Default: 2 ' aiSVG1_1
EmbedRasterImages	Boolean	If <code>true</code> , all raster images contained in the document are embedded in the saved file.
FontSubsetting	AiSVGFontSubsetting	The font glyphs included in the export file.
FontType	AiSVGFontType	The type of font included in the exported file. Default: 1 ' aiCEFFont
IncludeFileInfo	Boolean	If <code>true</code> , the XMP (Extensible Metadata Platform) metadata are included in the output file. Default: <code>false</code>
IncludeVariablesAndDatasets	Boolean	If <code>true</code> , Variables and Datasets are included. Default: <code>false</code>
OptimizeForSVGViewer	Boolean	If <code>true</code> , the Adobe namespace is included. Default: <code>false</code>

Property	Value type	What it is
PreserveEditability	Boolean	If <code>true</code> , preserve Illustrator editing capability when exporting the document. Default: <code>false</code>
Slices	Boolean	If <code>true</code> , preserve slice data in exported document. Default: <code>false</code>
SVGAutoKerning	Boolean	If <code>true</code> , SVG automatic kerning is allowed in the file. Default: <code>false</code>
SVGTextOnPath	Boolean	If <code>true</code> , the SVG text-on-path construct is allowed in the file. Default: <code>false</code>

► Exporting to SVG

```

'Exports the current document as SVG to dest
'dest contains the full path and file name to save to

Sub exportFileAsSVG (dest)
    Set appRef = CreateObject("Illustrator.Application")
    Set svgExportOptions = CreateObject("Illustrator.ExportOptionsSVG")

    If appRef.Documents.Count > 0 Then
        svgExportOptions.EmbedRasterImages = True
        svgExportOptions.FontSubsetting = 7 'aiAllGlyphs
        Set docRef = appRef.ActiveDocument
        Call docRef.Export (dest, 3, svgExportOptions) ' 3 = aiSVG
    End If
End Sub

```

FreeHandFileOptions

Options for opening a FreeHand file.

FreeHandFileOptions properties

Property	Value type	What it is
ConvertTextToOutlines	Boolean	If <code>true</code> , converts all text to outlines. Default: <code>false</code>
ImportSinglePage	Boolean	If <code>true</code> , imports only the page specified in the <code>PageToOpen</code> property. Default: <code>true</code>
PageToOpen	Long	The number of the page to import when opening a multipage document. Valid only when <code>ImportSinglePage</code> is <code>true</code> .

Gradient

A gradient definition contained in a document. A script can create new gradients.

Gradient properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
GradientStops	GradientStops collection object	Read-only. The gradient stops contained in this gradient.
Name	String	The gradient's name.
Parent	Document object	Read-only. The document that contains this gradient.
Type	AiGradientType	The kind of the gradient, either radial or linear.

Gradient methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.

► Creating and applying a gradient

```
'Creates a new gradient in the current document with 2
'gradient stops, then applies the gradient to the front-
'most path item
```

```
Set appRef = CreateObject("Illustrator.Application")
Set frontDocument = appRef.ActiveDocument
```

```
'Create a color for both ends of the gradient
Set startColor = CreateObject("Illustrator.RGBColor")
Set endColor = CreateObject("Illustrator.RGBColor")
```

```
startColor.Red = 0
startColor.Green = 100
startColor.Blue = 255
endColor.Red = 220
endColor.Green = 0
endColor.Blue = 100
```

```
'Create a new gradient
'new gradients always have 2 stops
Set newGradient = frontDocument.Gradients.Add
newGradient.Name = "Gradient created from script"
newGradient.Type = 1 'aiLinearGradient
```

```
'Modify the first gradient stop.
Set locationSpecification = newGradient.GradientStops(1)
```

```
locationSpecification.RampPoint = 30
locationSpecification.MidPoint = 60
locationSpecification.Color = startColor

'Modify the last gradient stop.
'The MidPoint for the last gradient stop is ignored
Set locationSpecification = newGradient.GradientStops(2)
locationSpecification.RampPoint = 80
locationSpecification.Color = endColor

'construct an Illustrator.GradientColor object referring to the
'newly created gradient
Set ColorOfGradient = CreateObject("Illustrator.GradientColor")
ColorOfGradient.Gradient = newGradient

'now get the frontmost path item and
'apply the new gradient as its fill
Set topPath = frontDocument.PathItems(1)
topPath.Filled = True
topPath.fillColor = ColorOfGradient
```

GradientColor

A gradient color specification, used in conjunction with the `Gradient` property of the `Color` specification.

A script can create a `GradientColor` using a reference to an existing gradient in the application. If no existing gradient object is referenced, a default gradient is supplied. An origin is used to specify the center point of the gradient in this specific gradient color. Single values are used to specify the gradient vector angles and lengths. A matrix may be specified to further transform the gradient color.

GradientColor properties

Property	Value type	What it is
Angle	Double	The gradient vector angle in degrees.
Application	Application object	Read-only. The <code>Illustrator Application</code> object.
Gradient	Gradient object	Reference to the object defining the gradient.
HiliteAngle	Double	The gradient hilite vector angle in degrees.
HiliteLength	Double	The gradient hilite vector length.
Length	Double	The gradient vector length.
Matrix	Matrix object	An additional transformation matrix to manipulate the gradient path.
Origin	Variant Array of 2 Doubles	The gradient vector origin.

► Changing a gradient color

```
'Creates a new RGB color then applies it as the color
'at the first gradient stop of the first gradient

Set appRef = CreateObject("Illustrator.Application")

Set frontDocument = appRef.Documents.Add(1) ' aiDocumentRGBColor

'Get a reference to the gradient that you want to change
Set firstGradient = frontDocument.Gradients(1)

'Create the new color
Set startColor = CreateObject("Illustrator.RGBColor")
startColor.Red = 0
startColor.Green = 255
startColor.Blue = 0

firstGradient.GradientStops(1).Color = startColor
```

Gradients

A collection of `Gradient` objects in a document.

Gradients properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Count	Long	Read-only. The number of objects in the collection.
Parent	Document object	Read-only. The parent document of this object.

Gradients methods

Method	Returns	What it does
Add ()	Gradient object	Creates a new object
Index (item as Gradient)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Gradient object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Removing a gradient

```
'Deletes the first gradient in the current document

Set appRef = CreateObject("Illustrator.Application")
Set frontDocument = appRef.ActiveDocument

'Get a reference to the gradient that you want to delete
Set gradientToDelete = frontDocument.Gradients(1)

'Now delete the gradient using the collection
frontDocument.Gradients.Remove gradientToDelete
```


GradientStop

A gradient stop definition contained in a specific gradient. Represents a point on a gradient that specifies a color change. For an example, see [GradientColor](#).

GradientStop properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Color	Color object	The color linked to this gradient stop.
Midpoint	Double	The distance between two GradientStops, in percentage. Range: 13.0 to 87.0
Parent	Document object	Read-only. The document that contains this gradient stop.
RampPoint	Double	The location of the color in the blend. Range: 0.0 to 100.0, where 100.0 is 100%

GradientStop methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.

GradientStops

A collection of GradientStop objects in a specific gradient.

GradientStops properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Count	Long	Read-only. The number of objects in the collection.
Parent	Document object	Read-only. The document that contains this object.

GradientStops methods

Method	Returns	What it does
Add ()	GradientStop object	Creates a new object
Index (item as GradientStop)	Long	Returns the index position of the object within the collection.
Item (itemKey)	GradientStops object	Returns an object reference to the object identified by itemKey (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Adding a new gradient stop

```
'Adds a new gray gradient stop to an existing gradient

Set appRef = CreateObject("Illustrator.Application")
Set frontDocument = appRef.ActiveDocument

'Get a reference to the gradient that you want to change
Set changeGradient = frontDocument.Gradients(1)

'Get a reference to the gradient stop that is the last one
'before you add a new gradient stop
originalCount = changeGradient.GradientStops.Count
Set lastStop = changeGradient.GradientStops(originalCount)

'add the new gradient stop
Set newStop = changeGradient.GradientStops.Add

'Set the values of the new gradient stop. move the original
'last gradient stop a bit to the left, and
'insert the new gradient stop at the old gradient stop's position
newStop.RampPoint = lastStop.RampPoint
lastStop.RampPoint = lastStop.RampPoint - 10

'Create a new color to apply to the newly created gradient stop
```

```
'choose a Gray tint value of 70%  
Set newStopColor = CreateObject("Illustrator.GrayColor")  
newStopColor.Gray = 70  
newStop.Color = newStopColor
```

GraphicStyle

Each graphic style defines a set of appearance attributes that you can apply to `PageItem` objects. Graphic styles are contained in documents. Scripts cannot create new graphic styles.

GraphicStyle properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Name	String	The <code>GraphicStyle</code> name.
Parent	Document object	Read-only. The document that contains this object.

GraphicStyle methods

Method	Returns	What it does
ApplyTo (artItem as PageItem)	Nothing	Applies the <code>GraphicStyle</code> to a specific art item.
MergeTo (artItem as PageItem)	Nothing	Merges this style into the current styles of a specified art item.
Delete ()	Nothing	Deletes the object.

► Applying a graphic style

This example duplicates and groups the current selection, applying the second graphic style in the document to the items in the group.

```
'Creates a new group item, adds a duplicate of each
'selected item to the group then applies a graphic style
'to the group

Set appRef = CreateObject("Illustrator.Application")

If appRef.Documents.Count > 0 Then
    If Not IsEmpty(appRef.Selection) Then
        mySelection = appRef.Selection
        endIndex = UBound(mySelection)
        startIndex = LBound(mySelection)
        Set newGroup = appRef.ActiveDocument.GroupItems.Add
        For i = startIndex To endIndex
            Set artItem = mySelection(i)
            Set dupItem = artItem.Duplicate(newGroup, aiPlaceAtEnd)
            appRef.ActiveDocument.GraphicStyles(2).ApplyTo dupItem
        Next
    End If
End If
```

GraphicStyles

A collection of graphic styles in a document.

GraphicStyles properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. The number of objects in the collection.
Parent	Document object	Read-only. The document that contains this object.

GraphicStyles methods

Method	Returns	What it does
Index (item as GraphicStyle)	Long	Returns the index position of the object within the collection.
Item (itemKey)	GraphicStyle object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Counting graphic styles

```
'Counts the number of graphic styles in the current document
'and stores in numStyles
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
If appRef.Documents.Count > 0 Then
    numStyles = appRef.ActiveDocument.GraphicStyles.Count
End If
```

GraphItem

A graph artwork item. Scripts cannot create new graph items.

GraphItem properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
ArtworkKnockout	AiKnockoutState	Is this GraphItem used to create a knockout, and if so, what kind.
BlendingMode	AiBlendModes	The mode used when compositing an object.
ContentVariable	Variable object	The content variable bound to this GraphItem. It is not necessary to set the type of the ContentVariable before binding. Illustrator automatically sets the type to AiGraph.
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Editable	Boolean	Read-only. If true, the GraphItem is editable.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the GraphItem excluding stroke width.
Height	Double	The height of the GraphItem, calculated on the GeometricBounds.
Hidden	Boolean	If true, this GraphItem is hidden.
IsIsolated	Boolean	If true, this GraphItem is isolated.
Layer	Layer object	Read-only. The Layer to which this GraphItem belongs.
Left	Double	The position of the left side of the GraphItem (in points, measured from the left side of the page).
Locked	Boolean	If true, this GraphItem is locked.
Name	String	The name of this GraphItem.
Note	String	The note assigned to this GraphItem.
Opacity	Double	The opacity of the GraphItem. Range: 0.0 to 100.0
Parent	Layer object or GroupItem object	Read-only. The parent of this GraphItem.
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
Selected	Boolean	If true, this GraphItem is selected.
Sliced	Boolean	If true, this GraphItem is sliced. Default: false

Property	Value type	What it is
Tags	Tags object	Read-only. The collection of <code>Tags</code> contained in this <code>GraphItem</code> .
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).
URL	String	The value of the Adobe URL tag assigned to this <code>GraphItem</code> .
VisibilityVariable	Variable	The visibility variable bound to this <code>GraphItem</code> . It is not necessary to set the type of the <code>VisibilityVariable</code> before binding. Illustrator automatically sets the type to <code>AiVisibility</code> .
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the <code>GraphItem</code> including stroke width.
Width	Double	The width of the <code>GraphItem</code> , based on the <code>GeometricBounds</code> .
WrapInside	Boolean	If <code>true</code> , the text frame object is wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If <code>true</code> , wrap text frame objects around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this <code>GraphItem</code> within the stacking order of the <code>GroupItem</code> or <code>Layer</code> (Parent) that contains the <code>GraphItem</code> .

GraphItem methods

Method	Returns	What it does
Copy ()	Nothing	Copies the <code>GraphItem</code> to the clipboard; the associated document must be the frontmost document.
Cut ()	Nothing	Cuts the <code>GraphItem</code> to the clipboard; the associated document must be the frontmost document.
Delete ()	Nothing	Deletes the object.
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	Object	Duplicates the art item, optionally specifying the location and position for the copy.

Method	Returns	What it does
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the art item, specifying the new location and position.
Resize (scaleX as Double , scaleY as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the <code>GraphItem</code> where <code>scaleX</code> is the horizontal scaling factor and <code>scaleY</code> is the vertical scaling factor. 100.0 = 100%
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the <code>GraphItem</code> relative to the current rotation; counter-clockwise if the <code>Angle</code> value is positive, clockwise if the value is negative.
Transform (transformationMatrix as Matrix , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the <code>GraphItem</code> by applying a transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the <code>GraphItem</code> relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the <code>GraphItem</code> 's position in the stacking order of the group or layer (Parent) of this object.

► Rotating graph items

```
'Rotates each graph item in the current document 90 degrees.
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
'Rotate each graph item 90 degrees
```

```
For Each graphRef In appRef.ActiveDocument.GraphItems
```

```
    graphRef.Rotate 90
```

```
Next
```

```
appRef.Redraw
```


GraphItems

A collection of `GraphItem` objects.

GraphItems properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. The number of objects in the collection.
Parent	Document object	Read-only. The document that contains this object.

GraphItems methods

Method	Returns	What it does
Index (item as GraphItem)	Long	Returns the index position of the object within the collection.
Item (itemKey)	<code>GraphItem</code> object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

GrayColor

A grayscale color specification used to apply a gray color to a layer or art item.

GrayColor properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Gray	Double	The tint of the gray. Range: 0.0 to 100.0, where 0.0 is black and 100.0 is white.

► Changing word color to gray

```
'Sets the fill color of the first word in the document to gray
```

```
Set appRef = CreateObject("Illustrator.Application")  
Set textColor = CreateObject("Illustrator.GrayColor")
```

```
'Get a reference to the first word in the active document  
Set text = appRef.ActiveDocument.TextFrames(1).TextRange  
Set firstWord = text.Words(1)
```

```
'Create the new color  
textColor.Gray = 45
```

```
firstWord.CharacterAttributes.fillColor = textColor
```

GroupItem

A grouped set of art items. Group items can contain all of the same page items that a layer can contain, including other nested groups.

Paths contained within a group or compound path in a document are returned as individual paths when a script asks for the paths contained in the document. However, paths contained in a group or compound path are not returned when a script asks for the paths in a layer which contains the group or compound path.

GroupItem properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
ArtworkKnockout	AiKnockoutState	Is this object used to create a knockout, and if so, what kind.
BlendingMode	AiBlendModes	The mode used when compositing an object.
Clipped	Boolean	If true, the GroupItem is clipped to the clipping mask.
CompoundPathItems	CompoundPathItems collection object	Read-only. The CompoundPathItems contained in this GroupItem.
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Editable	Boolean	Read-only. If true, this GroupItem is editable.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object excluding stroke width.
GraphItems	GraphItems collection object	Read-only. The raster items contained in this GroupItem.
GroupItems	GroupItems collection object	Read-only. The GroupItems contained in this GroupItem.
Height	Double	The height of the GroupItem, based on the GeometricBounds.
Hidden	Boolean	If true, this GroupItem is hidden.
IsIsolated	Boolean	If true, this object is isolated.
Layer	Layer object	Read-only. The layer to which this GroupItem belongs.
Left	Double	The position of the left side of the GroupItem (in points, measured from the left side of the page).
LegacyTextItems	LegacyTextItems	Read-only. The legacy text items in the group.
Locked	Boolean	If true, this GroupItem is locked.

Property	Value type	What it is
MeshItems	MeshItems collection object	Read-only. The MeshItems contained in this GroupItem.
Name	String	The name of this GroupItem.
Note	String	The note assigned to this GroupItem.
Opacity	Double	The opacity of the object. Range: 0.0 to 100.0
PageItems	PageItems collection object	Read-only. The PageItems contained in this GroupItem.
Parent	Document object	Read-only. The document that contains this GroupItem.
PathItems	PathItems collection object	Read-only. The PathItems contained in this GroupItem.
PlacedItems	PlacedItems collection object	Read-only. The PlacedItems contained in this GroupItem.
PluginItems	PluginItems collection object	Read-only. The PluginItems contained in this GroupItem.
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
RasterItems	RasterItem	Read-only. The RasterItems contained in this GroupItem.
Selected	Boolean	If true, this GroupItem is selected.
Sliced	Boolean	If true, this GroupItem is sliced. Default: false
SymbolItems	SymbolItems object	Read-only. The SymbolItems contained in this GroupItem.
Tags	Tags collection object	Read-only. The tags contained in this GroupItem.
TextFrames	TextFrames collection object	Read-only. The TextFrame objects contained in this GroupItem.
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).
URL	String	The value of the Adobe URL tag assigned to this GroupItem.
VisibilityVariable	Variable	The Variable bound to this GroupItem.
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the GroupItem including stroke width.
Width	Double	The width of the GroupItem, based on the GeometricBounds.

Property	Value type	What it is
WrapInside	Boolean	If <code>true</code> , the text frame object are wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If <code>true</code> , wrap text frame objects around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this group item within the stacking order of the group or layer (Parent) that contains this group item.

GroupItem methods

Method	Returns	What it does
Copy ()	Nothing	Copies the <code>GroupItem</code> to the clipboard; the associated document must be the frontmost document.
Cut ()	Nothing	Cuts the <code>GroupItem</code> to the clipboard; the associated document must be the frontmost document.
Delete ()	Nothing	Deletes the object.
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	Object	Duplicates the art item, optionally specifying the location and position for the copy.
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the art item, specifying the new location and position.
Paste ()	Nothing	Inserts the contents of the clipboard at the beginning of the <code>GroupItem</code> . You may only paste into a group that is contained in the active document.
Resize (scaleX as Double , scaleY as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the art item where <code>scaleX</code> is the horizontal scaling factor and <code>scaleY</code> is the vertical scaling factor; 100.0 = 100%

Method	Returns	What it does
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the art item relative to the current rotation. The object is rotated counter-clockwise if the <code>Angle</code> value is positive, clockwise if the value is negative.
Transform (transformationMatrix as Matrix [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the art item by applying a transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the art item relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the art item's position in the stacking order of the group or layer (Parent) of this object.

► Creating a group

It is easy to modify all of the objects contained in a group. This example demonstrates how to simplify your operations on multiple objects by creating a group to contain them.

```
'Creates a new group in the current document, containing
'a path item, and a text item

Set appRef = CreateObject("Illustrator.Application")

'Create a new group in the active document.
'This will be the group that holds the new triangle art
Set triangleGroup = appRef.ActiveDocument.GroupItems.Add

'Create a triangle and add text inside the group
Set trianglePath = triangleGroup.PathItems.Add
trianglePath.SetEntirePath Array(Array(100, 100), Array(300, 100),
Array(200, Tan(1.0471975) * 100 + 100))

Set captionText = triangleGroup.TextFrames.Add
captionText.Position = Array(100, 100)
captionText.Contents = "A triangle"
```

GroupItems

A collection of `GroupItem` objects.

GroupItems properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. The number of objects in the collection.
Parent	Document Object	Read-only. The document that contains this object.

GroupItems methods

Method	Returns	What it does
Add ()	GroupItem object	Creates a new object.
CreateFromFile (imagefile as String)	GroupItem object	Creates a <code>GroupItem</code> from a vector graphics file.
Index (item as GroupItem)	Long	Returns the index position of the object within the collection.
Item (itemkey)	GroupItem	Returns an object reference to the object identified by <code>itemkey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Importing a file using GroupItems

The following script shows how you can import a JPEG document using the `CreateFromFile` function. Before running this script you have to create a one-page PDF file and put it in the location.

`C:\temp\AiExport.jpg`.

```
'Adds a new group item to the document from a file (dest)
'dest contains the full path and file name of the file

Sub embedPDF (dest)
    Set appRef = CreateObject("Illustrator.Application")
    Set importedGroup = appRef.ActiveDocument.GroupItems.CreateFromFile
    (dest)
End Sub
```

IllustratorSaveOptions

Options that can be supplied when saving a document as an Illustrator file with the document `SaveAs` method.

IllustratorSaveOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Compatibility	AiCompatibility	Version of the Illustrator file format to create. Default: 13 ' aiIllustrator13
Compressed	Boolean	(Illustrator version 10 or later) If <code>true</code> , the saved file is compressed. Default: <code>true</code>
EmbedICCPProfile	Boolean	If <code>true</code> , an ICC profile is embedded in the saved file.
EmbedLinkedFiles	Boolean	If <code>true</code> , linked image files are included in the saved document (only valid for <code>SaveOptions</code> that specify an Illustrator compatibility of version 7 or later).
FlattenOutput	AiOutputFlattening	Specifies how transparency is flattened for file formats older than Illustrator 9.
FontSubsetThreshold	Double	Include a subset of fonts when less than this percentage of characters is used in the document (valid for Illustrator 9 or newer file format).
Overprint	AiPDFOverprint	The overprinting style. Default: 1 ' aiPreservePDFOverprint
PDFCompatible	Boolean	If <code>true</code> , save as a PDF compatible file (version 10 or later). Default: <code>true</code>

► Saving to Illustrator 8 format

This example illustrates how to save the frontmost document as Illustrator 8 format. Because the document is saved as a version earlier than 9, the example specifies how to convert opacity by breaking paths up in to sub-paths to preserve the appearance of the illustration.

```
'Saves current document in Illustrator format to dest with certain options
'specified in the script, dest contains the full path and file name to save
to
```

```
Sub exportfileToAi (dest)
    Set appRef = CreateObject("Illustrator.Application")
    Set saveOptions = CreateObject("Illustrator.IllustratorSaveOptions")
    saveOptions.Compatibility = 8'aiIllustrator8
    saveOptions.FlattenOutput = 1'aiPreserveAppearance
    appRef.Documents(1).SaveAs dest, saveOptions
End Sub
```


ImageCaptureOptions

Options for image capture, used with the [ImageCapture](#) method. All properties are optional.

ImageCaptureOptions properties

Property	Value type	What it is
AntiAliasing	Boolean	If <code>true</code> , the image result is anti-aliased. Default: <code>false</code>
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Matte	Boolean	If <code>true</code> , the artboard is matted with a color. Default: <code>false</code>
MatteColor	RGBColor object	The color to use for the artboard matte. Default: <code>white</code>
Resolution	Double	The resolution of the captured image file in points-per-inch (PPI), in the range 72.0 to 2400.0. Default: 150
Transparency	Boolean	If <code>true</code> , the image result is transparent. Default: <code>false</code>

Ink

Provides information about the ink name and related information.

Ink properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
InkInfo	InkInfo object	The ink information.
Name	String	The ink's name.

InkInfo

Specifies ink properties.

InkInfo properties

Property	Value type	What it is
Angle	Double	The ink's screen angle in degrees. Range: -360 to 360
Application	Application	Read-only. The Illustrator Application object.
CustomColor	Object	The color of the custom ink.
Density	Double	The neutral density. Minimum: 0.0
DotShape	String	The dot shape name.
Frequency	Double	The ink's frequency. Range: 0.0 to 1000.0
Kind	AiInkType	The ink type.
PrintingStatus	AiInkPrintStatus	The ink printing status.
Trapping	AiTrappingType	The trapping type.
TrappingOrder	Long	The order of trapping for the ink. Range: 1 to 4 for CMYK

► List inks in a document

```
'Gets the details of each ink in this document
'and displays them in a text frame
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
'create a new document and simple text frame
Set docRef = appRef.Documents.Add()
Set textRef = docRef.TextFrames.Add()
textRef.Top = docRef.Height - 100
textRef.Left = 100
```

```
'collect data from each Ink and InkInfo object
For Each inkRef In appRef.ActiveDocument.InkList
    sInks = sInks & inkRef.Name
    sInks = sInks & vbCrLf & vbTab
    sInks = sInks & "Frequency = " & inkRef.InkInfo.Frequency
    sInks = sInks & vbCrLf & vbTab
    sInks = sInks & "Density = " & inkRef.InkInfo.Density
    sInks = sInks & vbCrLf
Next
textRef.Contents = sInks
appRef.Redraw
```

InsertionPoint

A location between characters, used to insert new text objects. An insertion point is contained in an `InsertionPoints` collection. This is a [TextRange](#) object in which `CharacterOffset` indicates the location of the insertion point and `Length` is 0. This subclass does not define any additional properties.

InsertionPoints

A collection of `InsertionPoint` objects.

InsertionPoints properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	TextFrame object	Read-only. The object's container.

InsertionPoints methods

Method	Returns	What it does
Index (itemPtr as TextRange)	Long	Returns the index position of the object within the collection.
Item (itemKey)	InsertionPoint	Gets an element from the collection by <code>itemKey</code> (name or index).

► Using insertion point to add spaces

```
'Creates a new document, adds a text frame then
'uses insertion points to add a space between all characters
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
'create a new document and text frame
Set docRef = appRef.Documents.Add()
Set textRef = docRef.TextFrames.Add()
textRef.Contents = "Wouldn't you rather be scripting?"
textRef.Top = 400
textRef.Left = 100
textRef.TextRange.CharacterAttributes.Size = 20
appRef.Redraw
```

```
i = 2
Do While (i < textRef.InsertionPoints.Count)
    textRef.InsertionPoints(i).Characters.Add (" ")
    i = i + 2
Loop
```

LabColor

A color specification in the CIE Lab color space, used where a `color` object is required.

LabColor properties

Property	Value type	What it is
A	double	The a (red-green) color value. Range -128.0–128.0. Default: 0 . 0
B	double	The b (yellow-blue) color value. Range -128.0–128.0. Default: 0 . 0
L	double	The l (lightness) color value. Range -128.0–128.0. Default: 0 . 0
Name	string	Read-only. The class name of the referenced object.

Layer

A layer in an Illustrator document. Layers may contain nested layers, which are called sublayers in the user interface.

Layer object contains all of the page items in the layer as elements. Your script can access PageItems as elements of either the Layer object or as elements of the Document object. Only those page items in that layer can be accessed through the Layer object. To see page items in different layers, access them through the document.

Layer properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator Application object.
ArtworkKnockout	AiKnockoutState	Is this Layer used to create a knockout, and if so, what kind.
BlendingMode	AiBlendModes	The mode used when compositing an object.
Color	RGBColor object	The Layer's selection mark color.
CompoundPathItems	CompoundPathItems collection object	Read-only. The CompoundPathItems contained in this layer.
DimPlacedImages	Boolean	If true, placed images are rendered as dimmed in this layer.
GraphItems	GraphItems collection object	Read-only. The GraphItems collection contained in this layer.
GroupItems	GroupItems collection object	Read-only. The GroupItems contained in this layer.
HasSelectedArtwork	Boolean	If true, an object in this Layer has been selected; setting this property to false deselects all objects in the layer.
IsIsolated	Boolean	If true, this Layer is isolated.
Layers	Layers collection object	Read-only. The Layers contained in this layer.
LegacyTextItems	LegacyTextItems	Read-only. The legacy text items contained in this layer.
Locked	Boolean	If true, this layer is editable.
MeshItems	MeshItems collection object	Read-only. The MeshItems contained in this layer.
Name	String	The name of this layer.
Opacity	Double	The opacity of the layer. Range: 0.0 to 100.0
PageItems	PageItems collection object	Read-only. The PageItems contained in this layer.
Parent	Document object or Layer Object	Read-only. The document or Layer that contains this layer.

Property	Value type	What it is
PathItems	PathItems collection object	Read-only. The PathItems contained in this layer.
PlacedItems	PlacedItems collection object	Read-only. The PlacedItems contained in this layer.
PluginItems	PluginItems collection object	Read-only. The PluginItems contained in this layer.
Preview	Boolean	If true, this layer is displayed using preview mode.
Printable	Boolean	If true, this layer is printed when printing the document.
RasterItems	RasterItems collection object	Read-only. The RasterItems contained in this layer.
Sliced	Boolean	If true, this layer is sliced.
SymbolItems	SymbolItems collection object	Read-only. The SymbolItems contained in this layer.
TextFrames	TextFrameItems collection object	Read-only. The TextFrames contained in this layer.
Visible	Boolean	If true, this layer is visible.
ZOrderPosition	Long	Read-only. The position of this layer within the stacking order of Layers in the document

Layer methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the art item, specifying the new location and position. Does not change the position of the object on the art board, but changes the order in which Illustrator draws the objects, and the containment hierarchy. Places the object in the specified container, behind all other such objects.
Paste ()	Nothing	Pastes the contents of the clipboard into the Layer; if the associated document is the frontmost then all pasted objects remain selected after the paste.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the Layer's position in the stacking order of Layers in this document

► Count layers in a document

```
'Counts the number of layers in the current document  
'and stores in countOfLayers  
  
Set appRef = CreateObject("Illustrator.Application")  
  
'Get a reference to the layers, and obtain the total number  
Set frontDocument = appRef.ActiveDocument  
countOfLayers = frontDocument.Layers.Count
```


Layers

A collection of `Layer` objects.

Layers properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. The number of objects in the collection.
Parent	object	Read-only. The object that contains this <code>Layer</code> (can be another <code>Layer</code>).

Layers methods

Method	Returns	What it does
Add ()	<code>Layer</code> object	Creates a new object.
Index (item as <code>Layer</code>)	Long	Returns the index position of the object within the collection.
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Deleting layers

```
'Deletes all layers containing a specified string in their name
```

```
Set appRef = CreateObject("Illustrator.Application")
searchString = "Temp"
```

```
'Loop through all open documents
layersDeleted = 0
For Each targetDocument In appRef.Documents
    countOfLayers = targetDocument.Layers.Count
'Go through layers from the back to avoid changing the
'index of unvisited layers when we remove a layer
    For layerIndex = countOfLayers To 1 Step -1
        Set targetLayer = targetDocument.Layers(layerIndex)
        layerName = targetLayer.Name
        If (InStr(layerName, searchString) = 1) Then
            targetDocument.Layers.Remove targetLayer
            layersDeleted = layersDeleted + 1
        End If
    Next
Next
Next
```

LegacyTextItem

A text object created in Illustrator CS (version 10) or earlier, which is uneditable until converted. To convert legacy text, see [ConvertToNative](#).

You can view, move, and print legacy text, but you cannot edit it. Legacy text has an "x" through its bounding box when selected.

LegacyTextItem properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
ArtworkKnockout	AiKnockoutState	Is this object used to create a knockout, and if so, what kind.
BlendingMode	AiBlendModes	The mode used when compositing an object.
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Converted	Boolean	Read-only. If <code>true</code> , the legacy text has been updated to a native text frame item.
Editable	Boolean	Read-only. If <code>true</code> , this item is editable.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object excluding stroke width.
Height	Double	The height of the <code>LegacyTextItem</code> excluding stroke width, based on the <code>GeometricBounds</code> .
Hidden	Boolean	If <code>true</code> , this object is hidden.
IsIsolated	Boolean	If <code>true</code> , this object is isolated.
Layer	Layer object	Read-only. The layer to which this <code>LegacyTextItem</code> belongs.
Left	Double	The position of the left side of the item (in points, measured from the left side of the page).
Locked	Boolean	If <code>true</code> , this <code>LegacyTextItem</code> is locked.
Name	String	The name of this <code>LegacyTextItem</code> .
Note	String	The note assigned to this <code>LegacyTextItem</code> .
Opacity	Double	The opacity of the object. Range: 0.0 to 100.0
Parent	object	Read-only. The object that contains this <code>LegacyTextItem</code> .
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
Selected	Boolean	If <code>true</code> , this object is selected.

Property	Value type	What it is
Sliced	Boolean	If <code>true</code> , this <code>LegacyTextItem</code> is sliced.
Tags	Tags collection object	Read-only. The tags contained in this <code>LegacyTextItem</code> .
Top	Double	The top position of this <code>LegacyTextItem</code> .
URL	String	The value of the Adobe URL tag assigned to this <code>LegacyTextItem</code> .
VisibilityVariable	Variable	The <code>Variable</code> bound to this <code>LegacyTextItem</code> .
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the <code>LegacyTextItem</code> including stroke width.
Width	Double	The width of the <code>LegacyTextItem</code> excluding stroke width, based on the <code>GeometricBounds</code> .
WrapInside	Boolean	If <code>true</code> , the text frame object are wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If <code>true</code> , wrap text frame objects around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this art item within the stacking order of the group or layer (<code>Parent</code>) that contains the art item.

LegacyTextItem methods

Method	Returns	What it does
ConvertToNative ()	GroupItem Object	Converts the legacy text item to a text frame and deletes the original legacy text.
Copy ()	Nothing	Copies the legacy text item to the clipboard; the associated document must be the frontmost document.
Cut ()	Nothing	Cuts the legacy text item to the clipboard; the associated document must be the frontmost document.
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	Object	Duplicates the art item, optionally specifying the location and position for the copy.
Delete ()	Nothing	Deletes the object.

Method	Returns	What it does
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the art item, specifying the new location and position.
Resize (scaleX as Double , scaleY as Double [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the art item where <code>scaleX</code> is the horizontal scaling factor and <code>scaleY</code> is the vertical scaling factor; 100.0 = 100%
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the art item relative to the current rotation. The object is rotated counter-clockwise if the <code>Angle</code> value is positive, clockwise if the value is negative.
Transform (transformationMatrix as Matrix , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the art item by applying a transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the art item relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the art item's position in the stacking order of the group or layer (<code>Parent</code>) of this object.

LegacyTextItems

A collection of `LegacyText` items.

LegacyTextItems properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	object	Read-only. The object's container.

LegacyTextItems methods

Method	Returns	What it does
ConvertToNative ()	Boolean	Creates text frames from all legacy text items; the original legacy text items are deleted. Returns <code>true</code> if successful.
Index (itemPtr as LegacyTextItem)	Long	Returns the index position of the object within the collection.
Item (itemKey)	LegacyTextItem	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

Lines

A collection of `TextRange` objects corresponding to a line of text.

Lines properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	object	Read-only. The object's container.

Lines methods

Method	Returns	What it does
Index (itemPtr as TextRange)	Long	Returns the index position of the object within the collection.
Item (itemKey)	TextRange	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Counting lines

```
'Creates an area text item, adds some text, resizes
'the width of the path item containing the text item
'then counts the number of lines and stores in lineCount

'Create area text
textToPlace = "Scripting with Illustrator is fun and easy."
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()
Set itemRef = docRef.PathItems.Rectangle(500, 200, 50, 300)
Set textRef = docRef.TextFrames.AreaText(itemRef)
textRef.Contents = textToPlace
appRef.Redraw
'Change the path's width and display line count
itemRef.Width = 300
lineCount = textRef.Lines.Count
appRef.Redraw
```

Matrix

A transformation matrix specification, used to transform the geometry of objects.

Matrices are used in conjunction with the `Transform` method and as a property of a number of objects. You can generate an original matrix using the `Application` methods `getIdentityMatrix`, `getTranslationMatrix`, `getScaleMatrix`, or `getRotationMatrix`. If you need to apply multiple transformations to objects, it is more efficient to use the matrix suite than to apply the transformations one at a time.

A `Matrix` is a record containing the matrix values, not a reference to a matrix object. The matrix commands listed above operate on the values of a matrix record. If a command modifies a matrix, a modified matrix record is returned as the result of the command. The original matrix record passed to the command is not modified.

Matrix properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
MValueA	Double	Matrix property a.
MValueB	Double	Matrix property b.
MValueC	Double	Matrix property c.
MValueD	Double	Matrix property d.
MValueTX	Double	Matrix property tx.
MValueTY	Double	Matrix property ty.

► Applying transformations with a matrix

```
'Creates a new translation and rotation matrix then
'applies it to all items in the current document

Set appRef = CreateObject("Illustrator.Application")

'Move art half an inch to the right and 1.5 inch up on the page
Set moveMatrix = appRef.GetTranslationMatrix(72 * 0.5, 72 * 1.5)
'Add a rotation to the translation -- 10 degrees counterclockwise
Set totalMatrix = appRef.ConcatenateRotationMatrix(moveMatrix, 10)

'Apply the transformation to all art in the document
For Each artItem In appRef.ActiveDocument.PageItems
    artItem.Transform totalMatrix
Next
```

MeshItem

A gradient mesh art item. Mesh items cannot be created from a script, but can be copied and pasted.

MeshItem properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
ArtworkKnockout	AiKnockoutState	Is this object used to create a knockout, and if so, what kind.
BlendingMode	AiBlendModes	The mode used when compositing an object.
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Editable	Boolean	Read-only. If <code>true</code> , this <code>MeshItem</code> is editable.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object excluding stroke width.
Height	Double	The height of the <code>MeshItem</code> , based on the <code>GeometricBounds</code> .
Hidden	Boolean	If <code>true</code> , this <code>MeshItem</code> is hidden.
IsIsolated	Boolean	If <code>true</code> , this object is isolated.
Layer	Layer object	Read-only. The <code>Layer</code> to which this <code>MeshItem</code> belongs.
Left	Double	The position of the left side of the item (in points, measured from the left side of the page).
Locked	Boolean	If <code>true</code> , this <code>MeshItem</code> is locked.
Name	String	The name of this <code>MeshItem</code> .
Note	String	The note assigned to this <code>MeshItem</code> .
Opacity	Double	The opacity of the object. Range 0.0 to 100.0
Parent	object	Read-only. The object that contains this <code>MeshItem</code> .
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
Selected	Boolean	If <code>true</code> , this <code>MeshItem</code> is selected.
Sliced	Boolean	If <code>true</code> , this <code>MeshItem</code> is sliced.
Tags	Tags collection object	Read-only. The tags contained in this <code>MeshItem</code> .
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).

Property	Value type	What it is
URL	String	The value of the Adobe URL tag assigned to this <code>MeshItem</code> .
VisibilityVariable	Variant	The <code>Variable</code> bound to this <code>MeshItem</code> .
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the <code>MeshItem</code> including stroke width.
Width	Double	The width of the <code>MeshItem</code> , based on the <code>GeometricBounds</code> .
WrapInside	Boolean	If <code>true</code> , the text frame object are wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If <code>true</code> , wrap text frame objects around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this art item within the stacking order of the group or layer (<code>Parent</code>) that contains the art item.

MeshItem methods

Method	Returns	What it does
Copy ()	Nothing	Copies the mesh item to the clipboard; the associated document must be the frontmost document.
Cut ()	Nothing	Cuts the mesh item to the clipboard; the associated document must be the frontmost document.
Delete ()	Nothing	Deletes the object.
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	Object	Duplicates the art item, optionally specifying the location and position for the copy.
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the art item, specifying the new location and position.
Resize (scaleX as Double , scaleY as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the art item where <code>scaleX</code> is the horizontal scaling factor and <code>scaleY</code> is the vertical scaling factor; 100.0 = 100%

Method	Returns	What it does
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the art item relative to the current rotation. The object is rotated counter-clockwise if the <code>Angle</code> value is positive, clockwise if the value is negative.
Transform (transformationMatrix as Matrix , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the art item by applying a transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the art item relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the art item's position in the stacking order of the group or layer (<code>Parent</code>) of this object.

► Locking mesh items

```
'Locks all mesh items in the current document
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
For Each meshItem In appRef.ActiveDocument.MeshItems
    meshItem.Locked = True
Next
```

MeshItems

A collection of gradient mesh art items.

MeshItems properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	object	Read-only. The object's container.

MeshItems methods

Method	Returns	What it does
Index (itemPtr as MeshItem)	Long	Returns the index position of the object within the collection.
Item (itemKey)	MeshItem	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Copying mesh items between documents

To run this script, have two open documents:

- One document should contain at least one `MeshItem`.
- The other document can be empty. Make the empty document the frontmost before running the script.

```
'Copies all mesh items from one document
'to another, with an offset position
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
Set sourceDocument = appRef.Documents(1)
```

```
Set targetDocument = appRef.Documents.Add
```

```
locationOffset = 0
```

```
For Each meshItem In sourceDocument.MeshItems
```

```
    sourceDocument.Activate
```

```
    meshItem.Copy
```

```
    targetDocument.Activate
```

```
    targetDocument.Paste
```

```
'Get a reference to the item that was just copied into the document
```

```
    targetSelection = appRef.selection
```

```
    If (IsEmpty(targetSelection)) Then
```

```
        Exit For
```

```
    End If
```

```
    Set newMeshItem = targetSelection(0)
```

```
        newMeshItem.Position = Array(100, 400 + locationOffset)
        locationOffset = locationOffset + 50
    Next
```

NoColor

Represents the “none” color. Assignment of a reference to a `NoColor` object to the document’s default fill or stroke color, or those of an art item, is equivalent to setting their `Filled` or `Stroked` property to `False`.

NoColor properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator Application object.

► Removing a fill color

```
'Creates 2 overlapping, filled path items, then  
'sets the top item to "NoColor" to make the bottom item visible.
```

```
Set appRef = CreateObject("Illustrator.Application")  
Set docRef = appRef.Documents.Add()  
Set itemRef1 = docRef.PathItems.Rectangle(500, 200, 200, 100)  
Set itemRef2 = docRef.PathItems.Rectangle(550, 150, 200, 200)
```

```
Set colorRef = CreateObject("Illustrator.RGBColor")  
colorRef.Blue = 255  
itemRef1.FillColor = colorRef  
colorRef.Red = 255  
colorRef.Blue = 0  
itemRef2.FillColor = colorRef  
appRef.Redraw
```

```
'Create a nocolor object and assign it to the top object  
Set noColorRef = CreateObject("Illustrator.NoColor")  
itemRef2.FillColor = noColorRef  
appRef.Redraw
```

OpenOptions

Options to use when opening files in Illustrator.

OpenOptions properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
OpenAs	AiLibraryType	Opens the file as an Illustrator library.
UpdateLegacyGradientMesh	Boolean	If <code>true</code> , preserves the spot colors in the gradient mesh objects for legacy documents (pre-Illustrator CS3). Default: <code>true</code>
UpdateLegacyText	Boolean	If <code>true</code> , updates all text objects for documents saved with legacy text by Illustrator versions previous to CS. Default: <code>false</code>

► Opening a file and updating legacy text

This script assumes the file `C:\temp\LegacyText.ai` exists.

```
'Opens a file containing legacy text and updates the text automatically
'filePath contains the full path and file name of the file
```

```
Sub openOptions(filePath)
    Set appRef = CreateObject("Illustrator.Application")

    'Open a file with legacy text
    Set optionsRef = CreateObject("Illustrator.OpenOptions")
    optionsRef.UpdateLegacyText = True
    Set docRef = appRef.Open(filePath, 1, optionsRef) ' 1 =
aiDocumentRGBColor
End Sub
```

OpenOptionsAutoCAD

Options for opening an AutoCAD drawing, used with the [Open](#) method.

OpenOptionsAutoCAD properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
CenterArtwork	Boolean	If <code>true</code> , center the artwork that is created on the artboard. Default is <code>true</code> .
GlobalScaleOption	AiAutoCADGlobalScaleOption	How to scale the drawing on import. Default is <code>1 ' aiFitArtboard</code>
GlobalScalePercent	Double	The value when <code>globalScaleOption</code> is <code>2 ' aiScaleByValue</code> , expressed as a percentage. Range: 0.0 to 100.0 Default is <code>100.0</code>
ScaleLineweights	Boolean	If <code>true</code> , scales line weights by the same factor as the rest of the drawing. Default is <code>false</code> .
SelectedLayoutName	String	Name of the layout in the drawing to import.
Unit	AiAutoCADUnit	The unit to map to. Default is <code>3 ' aiMillimeters</code> .
UnitScaleRatio	Double	The ratio by which to scale while mapping units. Default is <code>1.0</code>

OpenOptionsPhotoshop

Options for opening an Adobe Photoshop document, used with the [Open](#) method.

OpenOptionsPhotoshop properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
LayerComp	String	The name of the layer comp to use when the document is converted.
PreserveHiddenLayers	Boolean	If <code>true</code> , preserve hidden layers when the document is converted. Default: <code>false</code> .
PreserveImageMaps	Boolean	If <code>true</code> , preserve image maps when the document is converted. Default: <code>true</code> .
PreserveLayers	Boolean	If <code>true</code> , preserve layers when the document is converted. Default: <code>true</code> .
PreserveSlices	Boolean	If <code>true</code> , preserve slices when the document is converted. Default: <code>true</code> .

PageItems

A collection of `PageItem` objects. Provides complete access to all the art items in an Illustrator document in the following classes:

```
CompoundPathItem
GraphItem
GroupItem
MeshItem
PathItem
PlacedItem
RasterItem
SymbolItem
TextFrame
```

You can reference page items through the `PageItems` property in a `Document`, `Layer`, or `Group`. When you access an individual item in one of these collections, the reference a page item of one of a particular type. For example, if you use `PageItems` to reference a graph item, the `typename` value of that object is `GraphItem`.

PageItems properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Object	Read-only. The object's container.

PageItems methods

Method	Returns	What it does
Index (itemPtr as PageItem)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Getting references to external files

Before running this script, open a document that contains one or more linked images.

```
'Gets the file references of all placed and raster items,
'then displays them in a new document
```

```
Set appRef = CreateObject("Illustrator.Application")
Dim fileReferences(9)
Set sourceDocument = appRef.ActiveDocument
index = 0
For Each artItem In sourceDocument.PageItems
```

```

        Select Case TypeName(artItem)
            Case "PlacedItem"
                fileReferences(index) = artItem.File
                index = index + 1
            Case "RasterItem"
                If (Not artItem.Embedded) Then
                    fileReferences(index) = artItem.File
                    index = index + 1
                End If
        End Select
    End Select
    If index = 10 Then
        Exit For
    End If
Next

'Write the file references to a new document
Set reportDocument = appRef.Documents.Add

Set fileNameText = reportDocument.TextFrames.Add
fileNameText.Position = Array(50, 520)
fileNameText.Contents = "File references in " & _
    sourceDocument.Name & ":"
For counter = 0 To (index - 1)
    Set fileNameText = reportDocument.TextFrames.Add
    fileNameText.Position = Array(65, 500 - 20 * counter)
    fileNameText.Contents = fileReferences(counter)
Next

```

► Getting page items of particular types

```

'Sets all placed or raster page items to hidden

Set appRef = CreateObject("Illustrator.Application")

For Each artItem In appRef.ActiveDocument.PageItems
    If (TypeName(artItem) = "PlacedItem" Or _
        TypeName(artItem) = "RasterItem") Then
        artItem.Hidden = True
    End If
Next

```

Paper

A container for information objects about the paper to be used for printing.

Paper properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Name	String	The paper name.
PaperInfo	PaperInfo object	The paper information.

PaperInfo

Contains information about the dimensions and imageable area of the paper to be used for printing.

PaperInfo properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
CustomPaper	Boolean	If <code>true</code> , a custom paper is being used.
Height	Double	The paper's height in points.
ImageableArea	Array of 4 Doubles	The imageable area.
Width	Double	The paper's width in points.

► Using Paper and PaperInfo objects

```
'Creates a new document then displays the paper list of the first printer in
a text frame
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
'Add a new document with some simple artwork
' (Must be a printable artwork for use with Printer object.)
Set docRef = appRef.Documents.Add()
Set textRef = docRef.TextFrames.Add()
textRef.Top = 750
textRef.Left = 50
```

```
curPrinterList = appRef.PrinterList
```

```
If UBound(curPrinterList) < 1 Then
    'exit
    sText = "Empty PrinterList"
    paragraphCount = 1
Else
    'Get paper objects for first printer
```

```
Set printerRef = curPrinterList(1)
sText = printerRef.Name
sText = sText & " paper list:" & vbCrLf

'Display data about available paper types
paragraphCount = 2
paperSizeArray = printerRef.PrinterInfo.PaperSizes
If IsEmpty (paperSizeArray) Then
    sText = sText & vbCrLf
    paragraphCount = paragraphCount + 1
Else
    For i = 0 To UBound(paperSizeArray) - 1
        Set paperRef = paperSizeArray(i)
        sText = sText & paperRef.Name & vbCrLf
        sText = sText & vbTab & paperRef.PaperInfo.Width
        sText = sText & " x " & paperRef.PaperInfo.Height & vbCrLf
        paragraphCount = paragraphCount + 2
    Next
End If
End If
textRef.Contents = sText
appRef.Redraw
}
```

ParagraphAttributes

Specifies the properties and attributes of a paragraph contained in a text frame.

Note: Paragraph attributes do not have default values, and are undefined until explicitly set.

ParagraphAttributes properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
AutoLeadingAmount	Double	Auto leading amount (as a percentage).
BunriKinshi	Boolean	If <code>true</code> , BunriKinshi is enabled.
BurasagariType	AiBurasagariTypeEnum	The Burasagari type, which specifies whether punctuation is allowed to fall outside of the paragraph bounding box (not available when Kinsoku Shori is set to <code>None</code>).
DesiredGlyphScaling	Double	The desired glyph scaling, expressed as a percentage of the default character width. Range: 50.0 to 200.0; at 100.0, the width of characters is not changed.
DesiredLetterSpacing	Double	Desired letter, spacing expressed as a percentage of the default kerning or tracking Range: -100.0 to 500.0; at 0, no space is added between letters; at 100.0, an entire space width is added between letters.
DesiredWordSpacing	Double	Desired word spacing, expressed as a percentage of the default space for the font. Range: 0.0 to 1000.0; at 100.00, no space is added between words.
EveryLineComposer	Boolean	If <code>true</code> , the Every-line Composer is enabled. If <code>false</code> , the Single-line Composer is enabled.
FirstLineIndent	Double	First line left indent (in points).
HyphenateCapitalizedWords	Boolean	If <code>true</code> , hyphenation is enabled for capitalized words.
Hyphenation	Boolean	If <code>true</code> , hyphenation is enabled for the paragraph.
HyphenationPreference	Double	Hyphenation preference scale for better spacing (0) or fewer hyphens (1). Range: 0.0 to 1.0

Property	Value type	What it is
HyphenationZone	Double	The distance (in points) from the right edge of the paragraph that marks the part of the line where hyphenation is not allowed. Note: 0 allows all hyphenation. Valid only when EveryLineComposer is false.
Justification	AiJustification	Paragraph justification.
Kinsoku	String	The name of a Kinsoku Shori set (a set of characters which cannot be used to begin or end a line of Japanese text).
KinsokuOrder	AiKinsokuOrderEnum	The preferred Kinsoku order.
KurikaeshiMojiShori	Boolean	If true, Kurikaeshi Moji Shori is enabled (controls how repeated characters are handled in Japanese text).
LeadingType	AiAutoLeadingType	The type of auto leading.
LeftIndent	Double	The left indent (in points).
MaximumConsecutiveHyphens	Long	Maximum number of consecutive hyphenated lines.
MaximumGlyphScaling	Double	Maximum glyph scaling, expressed as a percentage of the default character width. Range: 50.0 to 200.0; at 100.0, the width of characters is not changed. Note: Valid only for justified paragraphs.
MaximumLetterSpacing	Double	Maximum letter spacing, expressed as a percentage of the default kerning or tracking Range: -100.0 to 500.0; at 0, no space is added between letters; at 100.0, an entire space width is added between letters. Note: Valid only for justified paragraphs.
MaximumWordSpacing	Double	Maximum word spacing, expressed as a percentage of the default space for the font. Range: 0.0 to 1000.0; at 100.00, no space is added between words. Note: Valid only for justified paragraphs.
MinimumAfterHyphen	Long	Minimum number of characters after a hyphen.
MinimumBeforeHyphen	Long	Minimum number of characters before a hyphen.

Property	Value type	What it is
MinimumGlyphScaling	Double	Minimum glyph scaling, expressed as a percentage of the default character width. Range: 50.0 to 200.0; at 100.0, the width of characters is not changed. Note: Valid only for justified paragraphs.
MinimumHyphenatedWordSize	Long	Minimum number of characters for a word to be hyphenated.
MinimumLetterSpacing	Double	Minimum letter spacing, expressed as a percentage of the default kerning or tracking Range: -100.0 to 500.0; at 0, no space is added between letters; at 100.0, an entire space width is added between letters. Note: Valid only for justified paragraphs.
MinimumWordSpacing	Double	Minimum word spacing ,expressed as a percentage of the default space for the font. Range: 0.0 to 1000.0; at 100.00, no space is added between words.Note: Valid only for justified paragraphs.
Mojikumi	String	The name of a predefined Mojikumi set for Japanese text composition.
Parent	Object	Read-only. The object's container.
RightIndent	Double	Right indent of margin expressed in points.
RomanHanging	Boolean	If <code>true</code> , Roman hanging punctuation is enabled.
SingleWordJustification	AiJustification	Single word justification.
SpaceAfter	Double	Spacing after paragraph (in points).
SpaceBefore	Double	Spacing before paragraph (in points).
TabStops	Array of <code>TabStopInfo</code>	Tab stop settings, as <code>TabStopInfo</code> objects.

► Modifying paragraph attributes

```
'Creates a new document with an area text item
'containing 3 paragraphs then uses paragraph attributes
'to apply a different justification to each paragraph

Set appRef = CreateObject("Illustrator.Application")

'Create 3 new paragraphs
Set docRef = appRef.Documents.Add()
Set pathRef = docRef.PathItems.Rectangle(600, 200, 200, 400)
Set textRef = docRef.TextFrames.AreaText(pathRef)
textRef.Paragraphs.Add ("Left justified paragraph.")
textRef.Paragraphs.Add ("Center justified paragraph.")
textRef.Paragraphs.Add ("Right justified paragraph.")
textRef.TextRange.CharacterAttributes.Size = 28

'Change the justification of each paragraph
'using the paragraph attributes object
textRef.Paragraphs(1).ParagraphAttributes.Justification = 1 'aiRight
textRef.Paragraphs(2).ParagraphAttributes.Justification = 2 'aiCenter
textRef.Paragraphs(3).ParagraphAttributes.Justification = 0 'aiLeft
```


Paragraphs

A collection of Paragraph objects.

Paragraphs properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Object	Read-only. The object's container.

Paragraphs methods

Method	Returns	What it does
Add (contents as String [, relativeObject as TextFrame] [, insertionLocation as AiElementPlacement])	TextRange object	Adds a new paragraph with specified contents at the specified location in the document. If location is not specified, adds the paragraph to the containing text frame after the current text selection or insertion point.
AddBefore (contents as String)	TextRange object	Adds a new paragraph with specified contents before the current selection or insertion point.
Index (itemPtr as PageItem)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Object	Returns a reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Counting paragraphs in a document

```
'Counts the number of paragraphs in the document and
'stores in numParagraphs

Set appRef = CreateObject("Illustrator.Application")

If appRef.Documents.Count > 0 Then
    numParagraphs = 0
    For Each curTextArt In appRef.ActiveDocument.TextFrames
        Set curTextRange = curTextArt.TextRange
        numParagraphs = numParagraphs + curTextRange.Paragraphs.Count
    Next
End If
```

ParagraphStyle

Associates character and paragraph attributes with a style. The style object can be used to apply those attributes to the text in a `TextFrame` object.

ParagraphStyle properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
CharacterAttributes	CharacterAttributes	Read-only. The character properties for the text range.
Name	String	The <code>ParagraphStyle</code> 's name.
ParagraphAttributes	ParagraphAttributes	Read-only. The paragraph properties for the text range.
Parent	Object	Read-only. The object's container.

ParagraphStyle methods

Method	Returns	What it does
ApplyTo (textFrame as TextFrame [, clearingOverrides as Boolean])	Nothing	Applies the paragraph style to the text object.
Delete ()	Nothing	Deletes the object.

► Creating and applying a paragraph style

```
'Creates 3 paragraphs with different attributes, creates
'a paragraph style and applies it to each paragraph
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
'Create 3 new paragraphs
Set docRef = appRef.Documents.Add()
Set pathRef = docRef.PathItems.Rectangle(600, 200, 200, 400)
Set textRef = docRef.TextFrames.AreaText(pathRef)
textRef.Paragraphs.Add ("Left justified paragraph.")
textRef.Paragraphs.Add ("Center justified paragraph.")
textRef.Paragraphs.Add ("Right justified paragraph.")
textRef.TextRange.CharacterAttributes.Size = 28
```

```
'Change the justification of each paragraph
'using the paragraph attributes object
textRef.Paragraphs(1).ParagraphAttributes.Justification = 1 'aiRight
textRef.Paragraphs(2).ParagraphAttributes.Justification = 2 'aiCenter
textRef.Paragraphs(3).ParagraphAttributes.Justification = 0 'aiLeft
```

```
'Create a new paragraph style
```

```
Set paraStyle = docRef.ParagraphStyles.Add("LeftIndent")

'Add some paragraph attributes
paraStyle.ParagraphAttributes.Justification = 0 'aiLeft
paraStyle.ParagraphAttributes.FirstLineIndent = 10

For Each pg In textRef.Paragraphs
    paraStyle.ApplyTo pg, True
Next
appRef.Redraw
```

ParagraphStyles

A collection of ParagraphStyle objects. See [ParagraphStyle](#) for an example.

ParagraphStyles properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator Application object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Object	Read-only. The object's container.

ParagraphStyles methods

Method	Returns	What it does
Add (name as String)	ParagraphStyle object	Creates a new ParagraphStyle object.
Index (itemPtr as ParagraphStyle)	Long	Returns the index position of the object within the collection.
Item (itemKey)	ParagraphStyle object	Returns an object reference to the object identified by itemKey (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

PathItem

Art items such as those created using the Line, Rectangle, or Pen Tools. A path consists of path points that define its geometry. Path points are defined either as a `PathPoint` object or as an x-y page coordinate pair. The `PathItem` class gives you complete access to paths in Illustrator.

PathItem properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Area	Double	Read-only. The area of this path in square points. If the area is negative, the path is wound counterclockwise. Self-intersecting paths can contain sub-areas that cancel each other out, which makes this value zero even though the path has apparent area.
ArtworkKnockout	AiKnockoutState	Is this object used to create a knockout, and if so, what kind.
BlendingMode	AiBlendModes	The mode used when compositing an object.
Clipping	Boolean	If <code>true</code> , this path is to be used as a clipping path
Closed	Boolean	If <code>true</code> , this path is closed
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Editable	Boolean	Read-only. If <code>true</code> , this path item is editable.
Evenodd	Boolean	If <code>true</code> , the even-odd rule is used to determine insideness.
FillColor	Color object	The fill color of the path.
Filled	Boolean	If <code>true</code> , the path is filled.
FillOverprint	Boolean	If <code>true</code> , the art beneath a filled object are overprinted.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object excluding stroke width.
Guides	Boolean	If <code>true</code> , this path is a guide object.
Height	Double	The height of the path item, based on the <code>GeometricBounds</code> .
Hidden	Boolean	If <code>true</code> , this path item is hidden.
IsIsolated	Boolean	If <code>true</code> , this object is isolated.
Layer	Layer object	Read-only. The <code>Layer</code> to which this path item belongs.
Left	Double	The position of the left side of the item (in points, measured from the left side of the page).
Length	Double	Read-only. The length of this path (in points).

Property	Value type	What it is
Locked	Boolean	If <code>true</code> , this path item is locked.
Name	String	The name of this path item.
Note	String	The note text assigned to the path
Opacity	Double	The opacity of the object. Range 0.0 to 100.0
Parent	object	Read-only. The object that contains this path item.
PathPoints	PathPoints collection object	Read-only. The path points contained in this path item.
Polarity	AiPolarityValues	The polarity of the path.
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
Resolution	Double	The resolution of the path in dots per inch.
Selected	Boolean	If <code>true</code> , this path item is selected.
SelectedPathPoints	PathPoints collection object	Read-only. All of the selected path points in the path.
Sliced	Boolean	If <code>true</code> , this path item is sliced.
StrokeCap	AiStrokeCap	The type of line capping.
StrokeColor	Color object	The stroke color for the path.
Stroke	Boolean	If <code>true</code> , the path is stroked.
StrokeDashes	Variant Array of Doubles	The lengths for dashes and gaps in dashed lines, starting with the first dash length, followed by the first gap length, and so on. Set to an empty list, [], for a solid line.
StrokeDashOffset	Double	The default distance into the dash pattern at which the pattern is started.
StrokeJoin	AiStrokeJoin	Type of joints for the path.
StrokeMiterLimit	Double	When a default stroke join is set to <code>mitered</code> , this property specifies when the join will be converted to beveled (squared-off) by default. The default miter limit of 4 means that when the length of the point reaches four times the stroke weight, the join switches from a miter join to a bevel join. A value of 1 specifies a bevel join. Range: 1 to 500 Default: 4
StrokeOverprint	Boolean	If <code>true</code> , the art beneath a stroked object is overprinted.
StrokeWidth	Double	The width of the stroke (in points).
Tags	Tags collection object	Read-only. The tags contained in this path item.

Property	Value type	What it is
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).
URL	String	The value of the Adobe URL tag assigned to this path item.
VisibilityVariable	Variant	The <code>Variable</code> bound to this path item.
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the path item including stroke width.
Width	Double	The width of the path item, based on the <code>GeometricBounds</code> .
WrapInside	Boolean	If <code>true</code> , the text frame object is wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If <code>true</code> , wrap text frame objects around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this art item within the stacking order of the group or layer (<code>Parent</code>) that contains the art item.

PathItem methods

Method	Returns	What it does
Copy ()	Nothing	Copies the path item to the clipboard; the associated document must be the frontmost document.
Cut ()	Nothing	Cuts the path item to the clipboard; the associated document must be the frontmost document.
Delete ()	Nothing	Deletes the object.
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	Object	Duplicates the art item, optionally with the location and position for the copy.
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the art item to a new location and position.

Method	Returns	What it does
Resize (scaleX as Double , scaleY as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the art item where <code>scaleX</code> is the horizontal scaling factor and <code>scaleY</code> is the vertical scaling factor; 100.0 = 100%
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the art item relative to the current rotation. The object is rotated counter-clockwise if the <code>Angle</code> value is positive, clockwise if the value is negative.
SetEntirePath (pathSpecification as Array of [x, y] coordinate pairs)	Nothing	Sets the path using the specified array of (x,y) coordinate pairs for path points.
Transform (transformationMatrix as Matrix , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the art item by applying a transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the art item relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the art item's position in the stacking order of the group or layer (Parent) of this object.

► Setting attributes of a path

'Sets the fill color and stroke width of the first path item

```
Set appRef = CreateObject("Illustrator.Application")
```

```
Set frontDocument = appRef.ActiveDocument
Set firstPath = frontDocument.PathItems(1)
firstPath.Filled = True
firstPath.FillColor = frontDocument.Swatches(10).Color
firstPath.Stroked = True
firstPath.StrokeWidth = 5
```


► Setting a path consisting of straight lines

The `SetEntirePath` method provides an extremely efficient way to create paths that consist of straight lines. This script illustrates the use of the method.

```
'Creates 10 lines in the document then adds them as a single path item.
```

```
Set appRef = CreateObject("Illustrator.Application")  
Dim lineList(10)
```

```
For index = 0 To 10  
    lineList(index) = Array(index * 10 + 50, (index - 5) ^ 2 * 5 + 50)  
Next
```

```
Set frontDocument = appRef.ActiveDocument  
Set newPath = frontDocument.PathItems.Add  
newPath.SetEntirePath(lineList)
```

PathItems

A collection of `PathItem` objects. The methods `Ellipse`, `Polygon`, `Rectangle`, `RoundedRectangle`, and `Star` allow you to create complex `PathItems` using straightforward parameters. Note that `Rectangle`, `RoundedRectangle`, and `Ellipse` take the Y axis first (Top), then the X axis (Left). If you do not provide any parameters when calling these methods, default values are used.

PathItems properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Object	Read-only. The object's container.

PathItems methods

Method	Returns	What it does
Add ()	<code>PathItem</code> object	Creates a new <code>PathItem</code> object.
Ellipse ([top as Double] [, left as Double] [, width as Double] [, height as Double] [, reversed as Boolean] [, inscribed as Boolean])	<code>PathItem</code> object	Creates a new <code>PathItem</code> in the shape of an ellipse using the supplied parameters. Defaults: Top: 100 pt.; Left: 100 pPt.; Width: 50 pt.; Height: 100 pt.; Reversed: False
'Index (item as PathItem)	Long	Returns the index position of the object within the collection.
Item (itemKey)	<code>PathItem</code> object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
Polygon ([centerX as Double] [, centerY as Double] [, radius as Double] [, sides as Long] [, reversed as Boolean])	<code>PathItem</code> object	Creates a new <code>PathItem</code> in the shape of a polygon using the supplied parameters. Defaults: CenterX: 200 pt.; CenterY: 300 pt.; Radius: 50 pt.; Sides: 8; Reversed: false
Rectangle ([top as Double] [, left as Double] [, width as Double] [, height as Double] [, reversed as Boolean])	<code>PathItem</code> object	Creates a new <code>PathItem</code> in the shape of a rectangle using the supplied parameters.
RemoveAll ()	Nothing	Deletes all objects in this collection.

Method	Returns	What it does
RoundedRectangle ([top as Double] [, left as Double] [, width as Double] [, height as Double] [, horizontalRadius as Double] [, verticalRadius as Double] [, reversed as Boolean])	PathItem object	Creates a new PathItem in the shape of a rectangle with rounded corners using the supplied parameters. Defaults: HorizontalRadius: 15 pt.; VerticalRadius: 20 pt.; Reversed: False
Star ([centerX as Double] [, centerY as Double] [, radius as Double] [, innerRadius as Double] [, points as Long] [, reversed as Boolean])	PathItem object	Creates a new PathItem in the shape of a star using the supplied parameters. Defaults: CenterX: 200 pt.; CenterY: 300 pt.; Radius: 50 pt.; InnerRadius: 20 pt.; Points: 5; Reversed: False

► Creating shapes

```

'Creates 5 new path items in the current document and
'applies a random graphic style to each item

Set appRef = CreateObject("Illustrator.Application")

Set frontDocument = appRef.Documents.Add()

'Create a new rectangle with
'top = 762.5, left side = 87.5, width = 425 and height = 75
Set newRectangle = frontDocument.PathItems.Rectangle(762.5,87.5,425.0,75.0)
'create new rounded rectangle
Set newRndRectangle =
frontDocument.PathItems.RoundedRectangle(637.5,87.5,425.0,75.0,20.0,10.0)
'create ellipse, 'reversed' is false, 'inscribed' is true
Set newEllipse =
frontDocument.PathItems.Ellipse(512.5,87.5,425.0,75.0,false,true)
'create an 8 sided polygon
Set newOctagon = frontDocument.PathItems.Polygon(300.0,325.0,75.0,8)
'create a 4 pointed star
Set newStar = frontDocument.PathItems.Star(300.0,125.0,100.0,20.0,4)

For Each newItem in frontDocument.PathItems
    Randomize
    styleIndex = Int((frontDocument.GraphicStyles.Count) * Rnd + 1)
    frontDocument.GraphicStyles(styleIndex).ApplyTo(newItem)
Next

```

PathPoint

A point on a specific path. Each path point is made up of an anchor point and a pair of handles, or control points.

PathPoint properties

Property	Value type	What it is
Anchor	Variant Array of 2 Doubles	The position of the anchor point.
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
LeftDirection	Variant Array of 2 Doubles	The position of this path point's inward control point.
Parent	object	Read-only. The object that contains this <code>PathPoint</code> object.
PointType	AiPointType	The type of path point, either a curve or a corner. Any point can be considered a corner point. Setting the type to a corner forces the left and right direction points to be on a straight line when the user attempts to modify them in the user interface.
RightDirection	Variant Array of 2 Doubles	The position of this path point's outward control point.
Selected	AiPathPointSelection	Are points of this path point selected, and if so, which ones.

PathPoint methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.

► Shaping a path item by modifying a path point

```
'Changes the left and right path point directions of
'all but the last path point of frontmost path item

Set appRef = CreateObject("Illustrator.Application")
Set firstPath = appRef.ActiveDocument.PathItems(1)
countOfPoints = firstPath.PathPoints.Count

'Go through all PathPoints except the last one and set the
'left/right direction according to where the next point is
For index = 1 To (countOfPoints - 1)
    Set currentPoint = firstPath.PathPoints(index)
    Set nextPoint = firstPath.PathPoints(index + 1)
    currentAnchorList = currentPoint.Anchor
    nextAnchorList = nextPoint.Anchor
    deltax = nextAnchorList(0) - currentAnchorList(0)
```

```
    deltay = currentAnchorList(1) - nextAnchorList(1)
    length = Sqr(deltax ^ 2 + deltay ^ 2)

    currentPoint.LeftDirection = Array(currentAnchorList(0) - (50 * deltax /
length), currentAnchorList(1) - (50 * deltay / length))
    currentPoint.RightDirection = Array(currentAnchorList(0) + (50 * deltax /
length), currentAnchorList(1) + (50 * deltay / length))
Next
```

PathPoints

A collection of `PathPoint` objects in a path.

PathPoints properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Object	Read-only. The object's container.

PathPoints methods

Method	Returns	What it does
Add ()	<code>PathPoint</code> object	Creates a new <code>PathPoint</code> object.
Index (item as <code>PathPoint</code>)	Long	Returns the index position of the object within the collection.
Item (itemKey)	<code>PathPoint</code> object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Adding a new path point

```
'Creates a new path point in the first path item
```

```
Set appRef = CreateObject("Illustrator.Application")
Set firstPath = appRef.ActiveDocument.PathItems(1)
Set newPoint = firstPath.PathPoints.Add
```

```
newPoint.Anchor = Array(75, 300)
newPoint.LeftDirection = Array(10, 280)
newPoint.RightDirection = Array(165, 330)
newPoint.PointType = 2
```

Pattern

An Illustrator pattern definition contained in a document. Patterns are shown in the Swatches palette. Each pattern is referenced by a [PatternColor](#) object, which defines the pattern's appearance.

Pattern properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Name	String	The pattern name.
Parent	Document object	Read-only. The document that contains this pattern.

Pattern methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.

► Setting a fill color to a pattern

```
'Sets the default fill to pattern 1

Set appRef = CreateObject("Illustrator.Application")
Set frontDocument = appRef.Documents(1)
Set patternColor = CreateObject("Illustrator.PatternColor")

For Each swatchRef In appRef.ActiveDocument.Swatches
    ' Get the color object of the swatch
    Set swatchColor = swatchRef.Color
    ' Only operate on patterns
    If (swatchColor.TypeName = "PatternColor") Then
        frontDocument.DefaultFilled = true
        frontDocument.DefaultFillColor = swatchColor
    End If
Next
```

PatternColor

A pattern color specification, used in conjunction with the `Pattern` property of the `Color` specification. Pattern colors are created using a reference to an existing pattern in the document. A matrix may be specified to further transform the pattern color.

`PatternColor` objects can be used in any property that takes a color object, such as `fillColor` or `strokeColor`.

PatternColor properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Matrix	Matrix object	An additional transformation matrix to manipulate the prototype pattern.
Pattern	Pattern object	A reference to the pattern object to use in this color definition.
Reflect	Boolean	If <code>true</code> , the prototype is reflected before filling.
ReflectAngle	Double	The axis in degrees around which to reflect.
Rotation	Double	The angle in degrees to rotate the prototype pattern before filling.
ScaleFactor	Variant Array of 2 Doubles	The fraction to which to scale the prototype pattern before filling, represented as a point containing horizontal and vertical scaling percentages
ShearAngle	Double	The angle in degrees by which to slant the shear.
ShearAxis	Double	The axis in degrees to shear relative to.
ShiftAngle	Double	The angle in degrees to translate the unscaled prototype pattern before filling.
ShiftDistance	Double	The distance in points to which to translate the unscaled prototype pattern before filling.

► Modifying a pattern

```
'Rotates each pattern in the swatch list 10 degrees
'then applies it to the first path item.
```

```
Set appRef = CreateObject("Illustrator.Application")
Set colorOfPattern = CreateObject("Illustrator.PatternColor")
Set frontPath = appRef.ActiveDocument.PathItems(1)
```

```
For Each swatchRef In appRef.ActiveDocument.Swatches
    'Get the color object of the swatch
    Set swatchColor = swatchRef.Color
    'Only operate on patterns
    If (swatchColor.TypeName = "PatternColor") Then
        swatchColor.Rotation = 10
```



```
        frontPath.filled = true
        frontPath.fillColor = swatchColor
    End If
Next
```

Patterns

A collection of `Pattern` objects in a document.

Patterns properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Object	Read-only. The object's container.

Patterns methods

Method	Returns	What it does
Add ()	Pattern object	Creates a new <code>Pattern</code> object.
Index (item as <code>Pattern</code>)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Pattern object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Removing a pattern

```
'Deletes a pattern from the document
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
'Delete the pattern, then set the patternToRemove reference to  
'nothing because it no longer references an existing Illustrator pattern  
Set frontDocument = appRef.Documents(1)  
Set patternToRemove = frontDocument.Patterns(1)  
frontDocument.Patterns.Remove patternToRemove
```

PDFFileOptions

Options for opening Adobe PDF documents. This object is found in the `PDFFileOptions` property of the `Preferences` class.

PDFFileOptions properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
PageToOpen	Long	Specifies which page are used when opening a multipage document. Default: 1
Parent	Object	Read-only. The object's container.
PDFCropToBox	AiPDFBoxType	Specifies which box are used when placing a multipage document. Default: 4 ' AiPDFMediaBox

► Opening a PDF file

```
'Opens a multi-page PDF file to a specific  
'page using the PDFFileOptions object  
'filePath contains the full name and file path of the file to open
```

```
Sub pdfFileOptions(filePath)  
    Set appRef = CreateObject("Illustrator.Application")  
  
    With appRef.Preferences.PDFFileOptions  
        .PageToOpen = 2  
        .PDFCropToBox = 5 'aiPDFBoundingBox  
    End With  
  
    Set docRef = appRef.Open(filePath, 1)  
End Sub
```

PDFSaveOptions

Options that can be supplied when saving a document as an Adobe PDF file with the document `SaveAs` method.

PDFSaveOptions properties

Property	Value type	What it is
AcrobatLayers	Boolean	If <code>true</code> , create PDF layers from top-level layers (Acrobat 6 only option). Default: <code>false</code>
Application	Application object	Read-only. The Illustrator Application object.
BleedLink	Boolean	Link 4 bleed values. Default: <code>true</code>
BleedOffsetRect	Variant Array of 4 Doubles	The bleed offset rectangle.
ColorBars	Boolean	Draw color bars. Default: <code>false</code>
ColorCompression	AiCompressionQuality	Compression method for color bitmap images. Default: <code>1 ' aiNoCompression</code>
ColorConversionID	AiColorConversion	The PDF color conversion policy. Default: <code>0 ' aiColorConversionNone</code>
ColorDestinationID	AiColorDestination	The conversion target for color conversion. Default: <code>0 ' aiColorDestinationNone</code>
ColorDownsampling	Double	if zero, no downsampling, otherwise, the resolution to downsample color bitmap images to. Default: <code>150 . 0</code>
ColorDownsamplingImage-Threshold	Double	Downsample if the image's resolution is above this value. Default: <code>225 . 0</code>
ColorDownsamplingMethod	AiDownsampleMethod	Specifies how color bitmap images are resampled. Default: <code>0 ' aiNoDownsample</code>
ColorProfileID	AiColorProfile	The color profile to include. Default: <code>0 ' aiColorProfileNone</code>
ColorTileSize	Long	Tile size when compressing with JPEG2000. Default: <code>256</code>
Compatibility	AiPDFCompatibility	The version of the Acrobat® file format to create. Default: <code>5 ' Acrobat5</code>
CompressArt	Boolean	If <code>true</code> , line art and text are compressed. Default: <code>true</code>
DocumentPassword	String	A password string to open the document. Default: <code>no string</code>

Property	Value type	What it is
EnableAccess	Boolean	If <code>true</code> , enable accessing 128-bit. Default: <code>true</code>
EnableCopy	Boolean	If <code>true</code> , enable copying of text 128-bit. Default: <code>true</code>
EnableCopyAccess	Boolean	If <code>true</code> , enable copying and accessing 40-bit. Default: <code>true</code>
EnablePlainText	Boolean	Enable plaintext metadata 128-bit; available only for Acrobat 6. Default: <code>false</code>
FlattenerOptions	PrintFlattenerOptions object	The printing flattener options
FlattenerPreset	String	Transparency flattener style name. Default: <code>Custom</code>
FontSubsetThreshold	Double	Include a subset of fonts when less than this percentage of characters are used. Range: 0.0 to 100.0. Default: <code>100.0</code>
GenerateThumbnails	Boolean	If <code>true</code> , generate thumbnails for the saved document. Default: <code>true</code>
GrayscaleCompression	AiCompressionQuality	Specifies how grayscale bitmap images are compressed. Default: <code>1 ' aiNoCompression</code>
GrayscaleDownsampling	Double	If zero, no downsampling, otherwise, the resolution to downsample grayscale images to. Default: <code>150.0</code>
GrayscaleDownsampling-ImageThreshold	Double	Downsample if the image's resolution is above this value. Default: <code>225.0</code>
GrayscaleDownsampling-Method	AiDownsampleMethod	How should grayscale bitmap images be resampled. Default: <code>0 ' aiNoDownsample</code>
GrayscaleTileSize	Long	Tile size when compressing with JPEG2000. Default: <code>256</code>
MonochromeCompression	AiMonochromeCompression	How should monochrome bitmap images be compressed. Default: <code>0 ' aiNoMonoCompression</code>
MonochromeDownsampling	Double	The resolution to downsample images to. If 0, no downsampling. Default: <code>300.0</code>
MonochromeDownsampling-ImageThreshold	Double	Downsample if the image's resolution is above this value. Default: <code>450</code>

Property	Value type	What it is
MonochromeDownsampling-Method	AiDownsampleMethod	How monochrome bitmap images are resampled. Default: <code>aiNoDownsample</code>
Offset	Double	Offset from artwork to draw printer marks. Default: 0
Optimization	Boolean	If <code>true</code> , saved PDF are optimized for fast web viewing. Default: <code>false</code>
OutputCondition	String	An optional comment to add to the PDF file, describing the intended printing condition. Default: not included
OutputConditionID	String	The name of a registered printing condition. Default: not included
PageInformation	Boolean	If <code>true</code> , draw page information. Default: <code>false</code>
PageMarksType	AiPageMarksStyle	The page marks style. Default: <code>aiPageMarksStandard</code>
PDFAllowPrinting	AiPDFPrintAllowedEnum	PDF security printing permission. Default: 3 ' <code>aiPrint128HighResolution</code>
PDFChangesAllowed	AiPDFChangesAllowedEnum	PDF security changes allowed. Default: 5 ' <code>Change128AnyChanges</code>
PDFPreset	String	PDF preset name
PDFXStandard	AiPDFXStandard	The PDF standard with which this document complies. Default: 1 ' <code>aiPDFXNone</code>
PDFXStandardDescription	string	A description of the PDF standard from the selected preset.
PermissionPassword	String	A password string to restrict editing security settings. Default: no string
PreserveEditability	Boolean	If <code>true</code> , preserve Illustrator editing capabilities when saving the document. Default: <code>true</code>
PrinterResolution	Double	Flattening printer resolution. Default: 800.0
RegistrationMarks	Boolean	Draw registration marks. Default: <code>false</code>
RequireDocumentPassword	Boolean	If <code>true</code> , a password are required to open the document. Default: <code>false</code>
RequirePermissionPassword	Boolean	If <code>true</code> , a password are used to restrict editing security settings. Default: <code>false</code>

Property	Value type	What it is
Trapped	Boolean	If true, manual trapping has been prepared for the document. Default: false
TrimMarks	Boolean	If true, draw trim marks. Default: false
TrimMarkWeight	AiPDFTrimMarkWeight	Trim mark weight. Default: 1 ' aiTrimMarkWeight0125
ViewAfterSaving	Boolean	If true, view PDF after saving. Default: false

► Saving to PDF

This script illustrates how to save the frontmost document as PDF.

```
'Saves current document as PDF to dest
'dest contains the full path and file name to save to

Sub exportFileAsPDF (dest)
    Set appRef = CreateObject("Illustrator.Application")
    Set saveOptions = CreateObject("Illustrator.PDFSaveOptions")
    saveOptions.ColorCompression = 6 'aiJPEGHigh
    saveOptions.Compatibility = 5 'aiAcrobat5
    Set frontDocument = appRef.ActiveDocument
    Call frontDocument.SaveAs (dest, saveOptions)
End Sub
```

PhotoshopFileOptions

Options for opening a Photoshop file with the `Document` `open` method.

PhotoshopFileOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Parent	Object	Read only. This object's parent object.
PixelAspectRatioCorrection	Boolean	If <code>true</code> , imported images having non-square pixel aspect ratios are corrected.
PreserveImageMaps	Boolean	If <code>true</code> , the image maps are preserved when the document is converted. Default: <code>true</code>
PreserveLayers	Boolean	If <code>true</code> , layers are preserved when the document is converted. Default: <code>true</code>
PreserveSlices	Boolean	If <code>true</code> , slices are preserved when the document is converted. Default: <code>true</code>

► Opening a Photoshop document

```
'Opens a PSD file and retains its layers
'using the PhotoshopFileOptions object
'dest contains the full path and file name of the file to open
```

```
Sub photoShopFileOptions(dest)
    Set appRef = CreateObject("Illustrator.Application")

    With appRef.Preferences.PhotoshopFileOptions
        PreserveLayers = True
        PixelAspectRatioCorrection = False
    End With

    Set docRef = appRef.Open(dest, 1)
End Sub
```


PlacedItem

An artwork item placed in a document as a linked file. For example, an artwork item created using the File > Place command in Illustrator or by using the Add method of the PlacedItems collection object. For information, see [“PlacedItems” on page 141](#).

PlacedItem properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
ArtworkKnockout	AiKnockoutState	Is this object used to create a knockout, and if so, what kind.
BlendingMode	AiBlendModes	The mode used when compositing an object.
BoundingBox	Variant Array of 4 Doubles	Read-only. Dimensions of PlacedItem regardless of transformations.
ContentVariable	Variable	The Variable bound to this PlacedItem. It is not necessary to set the type of the ContentVariable before binding. Illustrator automatically sets the type to be the same as the PageItem to which it is bound.
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Editable	Boolean	Read-only. If true, this PlacedItem is editable.
File	String	The file containing the placed object.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object excluding stroke width.
Height	Double	The height of the PlacedItem, based on GeometricBounds.
Hidden	Boolean	If true, this PlacedItem is hidden.
IsIsolated	Boolean	If true, this object is isolated.
Layer	Layer object	Read-only. The layer this PlacedItem belongs to.
Left	Double	The position of the left side of the item (in points, measured from the left side of the page).
Locked	Boolean	If true, this PlacedItem is locked
Matrix	Matrix object	The transformation matrix applied to the PlacedItem.
Name	String	The name of this PlacedItem.
Note	String	The note assigned to this object.
Opacity	Double	The opacity of the object. Range: 0.0 to 100.0

Property	Value type	What it is
Parent	object	Read-only. The object that contains this <code>PlacedItem</code> .
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
Selected	Boolean	If <code>true</code> , this <code>PlacedItem</code> is selected.
Sliced	Boolean	If <code>true</code> , this <code>PlacedItem</code> is sliced.
Tags	Tags collection object	Read-only. The tags contained in this <code>PlacedItem</code> .
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).
URL	String	The value of the Adobe URL tag assigned to this <code>PlacedItem</code> .
VisibilityVariable	Variable	The Variable bound to this <code>PlacedItem</code> .
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the <code>PlacedItem</code> including stroke width.
Width	Double	The width of the <code>PlacedItem</code> , based on <code>GeometricBounds</code> .
WrapInside	Boolean	If <code>true</code> , the text frame object are wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If <code>true</code> , the text frame objects are wrapped around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this art item within the stacking order of the group or layer (<code>Parent</code>) that contains the art item.

PlacedItem methods

Method	Returns	What it does
Copy ()	Nothing	Copies the item to the clipboard; the associated document must be the frontmost document.
Cut ()	Nothing	Cuts the item to the clipboard; the associated document must be the frontmost document.
Delete ()	Nothing	Deletes the object.

Method	Returns	What it does
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	Object	Duplicates the art item, optionally with the location and position for the copy.
Embed ()	Nothing	Embeds this art in the document. Converts the art to art item objects as needed and deletes this object.
Resize (scaleX as Double , scaleY as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the art item where <code>scaleX</code> is the horizontal scaling factor and <code>scaleY</code> is the vertical scaling factor; 100.0 = 100%
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the art item relative to the current rotation. The object is rotated counter-clockwise if the <code>Angle</code> value is positive, clockwise if the value is negative.
Transform (transformationMatrix as Matrix , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the art item by applying a transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the art item relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the art item's position in the stacking order of the group or layer (<code>Parent</code>) of this object.

➤ **Changing the selection in placed items**

This script illustrates how to change the selection of PlacedItems.

```
'Toggles selection of all placed items in the document

Set appRef = CreateObject("Illustrator.Application")

For Each placedArt In appRef.ActiveDocument.PlacedItems
    placedArt.Selected = Not (placedArt.Selected)
Next
```

PlacedItems

A collection of placed art items. See [“PlacedItem” on page 137](#) for an example of use.

PlacedItems properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Document	Read-only. The object’s document container.

PlacedItems methods

Method	Returns	What it does
Add ()	PlacedItem object	Creates a new <code>PlacedItem</code> object. Use to place new art in a document. Use the <code>file</code> property of the resulting object to link the file containing the artwork. For information, see “PlacedItem” on page 137 .
Index (item as PlacedItem)	Long	Returns the index position of the object within the collection.
Item (itemKey)	PlacedItem object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

PluginItem

An art item created by an Illustrator plug-in such as the blend tool. Scripts can create a plugin item using `PlacedItem.trace` or `RasterItem.trace`, and can copy existing plugin items using the `duplicate` method, or copy and paste them, but cannot create `PluginItem` objects directly.

PluginItem properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
ArtworkKnockout	AiKnockoutState	Is this object used to create a knockout, and if so, what kind.
BlendingMode	AiBlendModes	The mode used when compositing an object.
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Editable	Boolean	Read-only. If <code>true</code> , this item is editable.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object excluding stroke width.
Height	Double	The height of the item, based on <code>GeometricBounds</code> .
Hidden	Boolean	If <code>true</code> , this item is hidden.
IsIsolated	Boolean	If <code>true</code> , this object is isolated.
IsTracing	Boolean	Read-only. If <code>true</code> , this plugin group represents a vector art item created by tracing a raster art item. The <code>tracing</code> property contains the tracing object associated with the options used to create it.
Layer	Layer object	Read-only. The layer this item belongs to.
Left	Double	The position of the left side of the item (in points, measured from the left side of the page).
Locked	Boolean	If <code>true</code> , this item is locked
Name	String	The name of this item.
Note	String	The note assigned to this <code>PluginItem</code> .
Opacity	Double	The opacity of the object. Range: 0.0 to 100.0
Parent	object	Read-only. The object that contains this item.
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
Selected	Boolean	If <code>true</code> , this item is selected.
Sliced	Boolean	If <code>true</code> , this item is sliced.

Property	Value type	What it is
Tags	Tags collection object	Read-only. The tags contained in this item.
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).
Tracing	TracingObject	Read-only. When this plugin group was created by tracing (<code>IsTracing</code> is <code>true</code>), the tracing object associated with the options used to create it.
URL	String	The value of the Adobe URL tag assigned to this item.
VisibilityVariable	Variable	The Variable bound to this item.
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the item including stroke width.
Width	Double	The width of the item, based on <code>GeometricBounds</code> .
WrapInside	Boolean	If <code>true</code> , the text frame object are wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If <code>true</code> , the text frame objects are wrapped around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this art item within the stacking order of the group or layer (<code>Parent</code>) that contains the art item.

PluginItem methods

Method	Returns	What it does
Copy ()	Nothing	Copies the item to the clipboard; the associated document must be the frontmost document.
Cut ()	Nothing	Cuts the item to the clipboard; the associated document must be the frontmost document.
Delete ()	Nothing	Deletes the object.
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	Object	Duplicates the art item, optionally with the location and position for the copy.
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the art item to a new location and position.

Method	Returns	What it does
Resize (scaleX as Double , scaleY as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the art item where <code>scaleX</code> is the horizontal scaling factor and <code>scaleY</code> is the vertical scaling factor; 100.0 = 100%
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the art item relative to the current rotation. The object is rotated counter-clockwise if the <code>Angle</code> value is positive, clockwise if the value is negative.
Trace ()	PluginItem object	Converts the raster art for this object to vector art, using default options. Reorders the placed art into the source art of a plugin group, and converts it into a group of filled and/or stroked paths that resemble the original image. Creates and returns a <code>PluginItem</code> object that references a <code>TracingObject</code> object.
Transform (transformationMatrix as Matrix , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the art item by applying a transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the art item relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the art item's position in the stacking order of the group or layer (Parent) of this object.

► Copying a plug-in item

This example demonstrates how to create a new `PluginItem` by copying an existing `PluginItem`.

```
'Copies and pastes the first plugin item in the current
'document

Set appRef = CreateObject("Illustrator.Application")

Set frontDocument = appRef.ActiveDocument
If (frontDocument.PluginItems.Count > 0) Then
    Set pluginArt = frontDocument.PluginItems(1)
    pluginArt.Copy
    frontDocument.Paste
Else
    MsgBox "There is no plug-in art in the front document"
End If
```

PluginItems

A collection of `PluginItem` objects in a document.

PluginItems properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Layer or <code>GroupItem</code> object	Read-only. The object's container.

PluginItems methods

Method	Returns	What it does
Index (item as <code>PluginItem</code>)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Plugin object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

PPDFile

Associates file information with a PostScript Printer Description (PPD) file.

PPDFile properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Name	String	The PPD model name.
PPDInfo	PPDFileInfo object	The PPD file information.

PPDFileInfo

Information about a PostScript Printer Description (PPD) file.

PPDFileInfo properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
LanguageLevel	String	The PostScript language level.
PPDFilePath	File object	Path specification for the PPD file.
ScreenList	Variant	List of color separation screens.
ScreenSpotFunctionList	Variant	List of color separation screen spot functions.

► Using a PPD file

```
'Displays the name, postscript level and path data contained in
'each PPD file in a text frame
```

```
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()
x = 30
y = docRef.Height - 30
```

```
For Each ppd In appRef.PPDFileList
    ' get data for each PPD
    sPPD = ppd.Name & vbCrLf
    sPPD = sPPD & vbTab & "PS Level "
    sPPD = sPPD & ppd.PPDInfo.LanguageLevel & vbCrLf
    sPPD = sPPD & vbTab & "Path: "
    sPPD = sPPD & ppd.PPDInfo.File & vbCrLf

    ' display data with a textFrame
    Set textRef = docRef.TextFrames.Add()
    textRef.TextRange.CharacterAttributes.Size = 8
```

```
    textRef.Contents = sPPD
    textRef.Top = y
    textRef.Left = x
    appRef.Redraw
    y = y - textRef.Height
Next
```

Preferences

Specifies the preferred options for AutoCAD, FreeHand, PDF, and Photoshop files.

Preferences properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
AutoCADFileOptions	OpenOptionsAutoCAD object	Read-only. Options to use when opening or placing an AutoCAD file.
FreeHandFileOptions	FreeHandOpenOptions object	Read-only. Options to use when opening or placing a FreeHand file.
Parent	Object	Read-only. The object's container.
PDFFileOptions	PDFFileOptions	Read-only. Options to use when opening or placing a PDF file.
PhotoshopFileOptions	PhotoshopFileOptions	Read-only. Options to use when opening or placing a Photoshop file.

Preferences methods

Method	Returns	What it does
GetBooleanPreference (key as String)	Boolean	Gets the boolean value of a given application preference.
GetIntegerPreference (key as String)	Integer	Gets the integer value of a given application preference.
GetRealPreference (key as String)	Double	Gets the real-number value of a given application preference.
GetStringPreference (key as String)	String	Gets the string value of a given application preference.
RemovePreference (key as String)	Nothing	Deletes a given application preference.
SetBooleanPreference (key as String , value as Boolean)	Nothing	Sets the boolean value of a given application preference.
SetIntegerPreference (key as String , value as Integer)	Nothing	Sets the integer value of a given application preference.

Method	Returns	What it does
SetRealPreference (key as String , value as Double)	Nothing	Sets the real-number value of a given application preference.
SetStringPreference (key as String , value as String)	Nothing	Sets the string value of a given application preference.

PrintColorManagementOptions

Contains information used for color management of the document.

PrintColorManagementOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
ColorProfileMode	AiPrintColorProfile	The color management profile mode. Default: 1 ' aiSourceProfile
Intent	AiPrintColorIntent	The color management intent type. Default: 2 ' aiRelativeColorimetric
Name	String	The color management profile name.

► Managing colors for printing

```
'Creates a new document with a path item,
'creates a PrintColorManagementOptions object and assigns it
'to a PrintOptions object, then prints with each color intent
```

```
'create a simple path item and apply a graphic style to it
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add
Set pathItem = docRef.PathItems.Rectangle(600, 200, 200, 200)
docRef.GraphicStyles(2).ApplyTo pathItem
```

```
Set colorOptions = CreateObject("Illustrator.PrintColorManagementOptions")
Set printOptionsRef = CreateObject("Illustrator.PrintOptions")
printOptionsRef.ColorManagementOptions = colorOptions
colorOptions.Name = "ColorMatch RGB"
```

```
'Print the current document once for each color intent.
colorOptions.Intent = 3 'aiAbsoluteColorimetric
docRef.print
colorOptions.Intent = 0 'aiPerceptualIntent
docRef.print
colorOptions.Intent = 2 'aiRelativeColorimetric
docRef.print
colorOptions.Intent = 1 'aiSaturationIntent
docRef.print
```

PrintColorSeparationOptions

Information about the color separations to be used in printing the document.

PrintColorSeparationOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
ColorSeparationMode	AiPrintColorSeparationMode	The color separation type. Default: 0 'aiComposite
ConvertSpotColors	Boolean	If true, spot colors are converted to process colors. Default: false
InkList	Variant Array of Ink objects	The list of inks for color separation.
OverPrintBlack	Boolean	If true, overprint in black. Default: false

► Managing print color separations

```
'Creates a new document, adds symbol items, create
'a PrintColorSeparationOptions object and
'print with various separation settings

'Create a new document and add some artwork
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()
y = docRef.Height - 30
i = 1
Do While (i < 2)
    Set symbolRef = docRef.Symbols(i)
    Set itemRef = docRef.SymbolItems.Add(symbolRef)
    itemRef.Top = y
    itemRef.Left = 100
    y = (y - (itemRef.Height + 10))
    i = i + 1
Loop
appRef.Redraw

'Create a separations object and assign it a
'PrintOptions object
Set printOpts = CreateObject("Illustrator.printOptions")
Set separationOpts =
CreateObject("Illustrator.PrintColorSeparationOptions")
printOpts.ColorSeparationOptions = separationOpts

'Print with various separation options
separationOpts.ConvertSpotColors = True
separationOpts.OverPrintBlack = True
separationOpts.ColorSeparationMode = 0 'aiComposite
docRef.PrintOut printOpts
```



```
separationOpts.ColorSeparationMode = 2 'aiInRIPSeparation  
docRef.PrintOut printOpts
```

```
separationOpts.ConvertSpotColors = False  
separationOpts.OverPrintBlack = False  
separationOpts.ColorSeparationMode = 1 'aiHostBasedSeparation  
docRef.PrintOut printOpts
```

PrintCoordinateOptions

Information about the media and associated printing parameters.

PrintCoordinateOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Emulsion	Boolean	If true, flip artwork will be flipped horizontally. Default: false
FitToPage	Boolean	Whether to proportionally scale the artwork to fit on media. Default: false
HorizontalScale	Double	The horizontal scaling factor expressed as a percentage (100 = 100%) Range: 1.0 to 10000.0. Default: 100.0
Orientation	AiPrintOrientation	The artwork orientation. Default: 0 ' aiPortrait
Position	AiPrintPosition	The artwork position on media. Default: 5 ' aiTranslateCenter
Tiling	AiPrintTiling	The page tiling mode. Default: 0 ' aiTileSingleFullPage
VerticalScale	Double	The vertical scaling factor expressed as a percentage (100 = 100%) Range: 1.0 to 10000.0. Default: 100.0

► Managing print coordinates

```
'Creates a new document with text extending off the page
'and print it with various Coordinate Options
```

```
'Create a TextFrame that extends off the page
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()
Set textRef = docRef.TextFrames.Add()
textRef.Contents = "This should extend off the page"
textRef.Left = -50
textRef.Top = 600
textRef.Width = docRef.Width + 100
textRef.Height = 150
appRef.Redraw

'Print the item using various settings of the
'PrintCoordinateOptions object
Set coordinateOptions = CreateObject("Illustrator.PrintCoordinateOptions")
Set printOptions = CreateObject("Illustrator.printOptions")
printOptions.coordinateOptions = coordinateOptions

coordinateOptions.Emulsion = True ' reverse from right to left
coordinateOptions.FitToPage = True ' fit artwork to page size
coordinateOptions.Orientation = 1 'aiLandscape
docRef.PrintOut printOptions
```

```
coordinateOptions.Emulsion = False  
coordinateOptions.Orientation = 0 'aiPortrait  
coordinateOptions.HorizontalScale = 50  
coordinateOptions.VerticalScale = 50  
docRef.PrintOut printOptions
```

Printer

Associates an available printer with printer information. To request a list of printers, you must first have a document open or an error is returned.

Printer properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Name	String	The printer name
PrinterInfo	PrinterInfo object	The printer information

PrinterInfo

Contains all configuration information about a printer.

PrinterInfo properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
BinaryPrintingSupport	Boolean	If <code>true</code> , the printer supports binary printing.
ColorSupport	AiPrinterColorMode	The printer color capability.
CustomPaperSupport	Boolean	If <code>true</code> , the printer supports custom paper sizes.
CustomPaperTransverse-Support	Boolean	If <code>true</code> , the printer supports custom paper transverse.
DeviceResolution	Double	The printer default resolution.
InRIPSeparationSupport	Boolean	If <code>true</code> , the printer supports InRIP color separation.
MaxDeviceResolution	Double	The printer maximum device resolution.
MaxPaperHeight	Double	Custom paper's maximum height.
MaxPaperHeightOffset	Double	Custom paper's maximum height offset.
MaxPaperWidth	Double	Custom paper's maximum width.
MaxPaperWidthOffset	Double	Custom paper's maximum width offset.
MinPaperHeight	Double	Custom paper's minimum height.
MinPaperHeightOffset	Double	Custom paper's minimum height offset.
MinPaperWidth	Double	Custom paper's minimum width.
MinPaperWidthOffset	Double	Custom paper's minimum width offset.

Property	Value type	What it is
PaperSizes	Variant Array of Paper objects	The list of supported paper sizes.
PostScriptLevel	AiPrinterPostScriptLevelEnum	The PostScript language level.
PrinterType	AiPrinterTypeEnum	The printer type.

► Finding printers

```
'Uses the PrinterList to obtain the name
'of each printer and displays it in a text frame

noFailure = True
Set appRef = CreateObject("Illustrator.Application")

Set docRef = appRef.Documents.Add()
Set textRef = docRef.TextFrames.Add()
textRef.Top = docRef.Height - 50
textRef.Left = 50
sData = "Printers:"

For Each printerRef In appRef.PrinterList
    sData = sData & vbCrLf & printerRef.Name
Next

textRef.Contents = sData
appRef.Redraw
```

► Finding printer information

```
'Creates a document with a text frame then
'loops through the printer list, showing info on
'each printer using the PrintInfo attribute

'create a new document
Set appRef = CreateObject("Illustrator.Application")

Set docRef = appRef.Documents.Add()

'add title text frame
Set textRef1 = docRef.TextFrames.Add()
textRef1.Contents = "Checking Printers..."
textRef1.Top = 600
textRef1.Left = 200
appRef.Redraw

'for each printer, list the PS support and InHostRip support
For Each printerRef In appRef.PrinterList
    sPrintInfo = sPrintInfo & printerRef.Name & vbCrLf
    sPrintInfo = sPrintInfo & vbTab & "PS Level = "
    sPrintInfo = sPrintInfo & CStr(printerRef.PrinterInfo.PostScriptLevel) & vbCrLf
    sPrintInfo = sPrintInfo & vbTab & "Device Resolution = "
    sPrintInfo = sPrintInfo & CStr(printerRef.PrinterInfo.DeviceResolution) &
```

```
        & vbCrLf
        sPrintInfo = sPrintInfo & vbTab & "InRIPSeparation Support = "
        sPrintInfo = sPrintInfo &
        CStr(printerRef.PrinterInfo.InRIPSeparationSupport) & vbCrLf
    Next

    textRef1.Contents = sPrintInfo
    appRef.Redraw
```

PrintFlattenerOptions

Contains flattening options for use when Illustrator outputs artwork that contains transparency into a non-native format.

PrintFlattenerOptions properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
ClipComplexRegions	Boolean	If <code>true</code> , clip complex regions. Default: <code>false</code>
ConvertStrokesToOutlines	Boolean	If <code>true</code> , convert all strokes to outlines. Default: <code>false</code>
ConvertTextToOutlines	Boolean	If <code>true</code> , convert all text to outlines. Default: <code>false</code>
FlatteningBalance	Double	The flattening balance. Range: 0.0 to 100.0. Default: 100.0
GradientResolution	Long	The gradient resolution in dots per inch (dpi). Range: 1.0 to 9600.0. Default: 300.0
Overprint	AiPDFOverprint	Whether to preserve, discard, or simulate overprinting. Default: 1 ' aiPreservePDFOverprint
RasterizationResolution	Double	The rasterization resolution in dots per inch (dpi). Range: 1.0 to 9600.0. Default: 300.0

► Setting print flattening

```
'Creates a document, adds a rectangle
'with a graphic style applied, then
'prints the document with "low" and "high"
'flattener settings

Set appRef = CreateObject("Illustrator.Application")

Set docRef = appRef.Documents.Add()
Set itemRef = docRef.PathItems.Rectangle(600, 200, 200, 200)
docRef.GraphicStyles(2).ApplyTo itemRef

'Create a PrintFlattenerOptions object and
'assign it to a PrintOptions object.
Set flatOpts = CreateObject("Illustrator.PrintFlattenerOptions")
Set printOpts = CreateObject("Illustrator.printOptions")
printOpts.FlattenerOptions = flatOpts

'print faster with low resolution
flatOpts.ClipComplexRegions = True
flatOpts.GradientResolution = 30
flatOpts.RasterizationResolution = 30
docRef.PrintOut printOpts

'print slower with higher resolution
flatOpts.ClipComplexRegions = False
```

```
flatOpts.GradientResolution = 300  
flatOpts.RasterizationResolution = 300  
docRef.PrintOut printOpts
```


PrintFontOptions

Contains information about font downloading and substitution for the fonts used for printing a document.

PrintFontOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
DownloadFonts	AiPrintFontDownloadMode	The font download mode. Default: 1 ' aiDownloadSubset
FontSubstitution	AiFontSubstitutionPolicy	The font substitution policy. Default: 1 ' aiSubstituteOblique

► Setting print font options

```
'Creates a new document with a text item,
'creates a new print font options object then
'prints with specified font options

Set appRef = CreateObject("Illustrator.Application")

Set docRef = appRef.Documents.Add()
Set textRef = docRef.TextFrames.Add()
textRef.Top = 600
textRef.Left = 50
textRef.Contents = "PrintFontOptions object"

'Create a PrintFontOptions object and
'assign it to a PrintOptions object.
Set fontOpts = CreateObject("Illustrator.PrintFontOptions")
Set printOpts = CreateObject("Illustrator.printOptions")
printOpts.FontOptions = fontOpts

fontOpts.DownloadFonts = 2 'aiDownloadComplete
fontOpts.FontSubstitution = 0 'aiSubstituteOblique

'print it
docRef.PrintOut printOpts
```

PrintJobOptions

Contains information about how a job is to be printed.

PrintJobOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
BitmapResolution	Double	The bitmap resolution. Minimum 0 . 0. Default: 0 . 0
Collate	Boolean	If true, collate print pages. Default: false
Copies	Long	The number of copies to print. Minimum: 1 Default: 1
Designation	AiPrintArtworkDesignation	The layers/objects to be printed. Default: 0 ' aiVisiblePrintableLayers
File	String	The file to which to print.
Name	String	The print job name.
PrintArea	AiPrintingBounds	The printing bounds. Default: 0 ' aiArtboardBounds
PrintAsBitmap	Boolean	If true, print as bitmap. Default: false
ReversePages	Boolean	If true, print pages in reverse order. Default: false

► Printing with job options

```
'Creates a new document containing text in visible,
'printable, non-visible and non-printable layers then
'prints document with different designations to demonstrate
'how each designation affects which layers are printed
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
Set docRef = appRef.Documents.Add()
```

```
'create a visible, printable item
```

```
Set textRef_1 = docRef.Layers(1).TextFrames.Add()
```

```
textRef_1.Contents = "Visible and Printable"
```

```
textRef_1.Top = 600
```

```
textRef_1.Left = 200
```

```
'create a visible, non-printable item
```

```
Set layerRef_2 = docRef.Layers.Add()
```

```
Set textRef_2 = layerRef_2.TextFrames.Add()
```

```
textRef_2.Contents = "Visible and Non-Printable"
```

```
textRef_2.Top = 500
```

```
textRef_2.Left = 250
```

```
layerRef_2.Printable = False
```

```
' create a non-visible item
Set layerRef_3 = docRef.Layers.Add()
Set textRef_3 = layerRef_3.TextFrames.Add()
textRef_3.Contents = "Non-Visible"
textRef_3.Top = 400
textRef_3.Left = 300
layerRef_3.Visible = False
appRef.Redraw

' Print with various jobOptions
Set jobOptionsRef = CreateObject("Illustrator.PrintJobOptions")
Set printOpts = CreateObject("Illustrator.printOptions")
printOpts.JobOptions = jobOptionsRef

jobOptionsRef.Designation = 2 'aiAllLayers
jobOptionsRef.ReversePages = True
docRef.PrintOut printOpts

jobOptionsRef.Designation = 1 'aiVisibleLayers
jobOptionsRef.ReversePages = False
jobOptionsRef.PrintAsBitmap = True
docRef.PrintOut printOpts

jobOptionsRef.Designation = 0 'aiVisiblePrintableLayers
jobOptionsRef.File = "C:\temp\printJobTest1.ps"
docRef.PrintOut printOpts
```

PrintOptions

Collects all information about all printing options including flattening, color management, coordinates, fonts, and paper. Used as an argument to the [PrintOut](#) method.

PrintOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
ColorManagementOptions	PrintColorManagementOptions object	The printing color management options.
ColorSeparationOptions	PrintColorSeparationOptions object	The printing color separation options.
CoordinateOptions	PrintCoordinateOptions object	The printing coordinate options.
FlattenerOptions	PrintFlattenerOptions object	The printing flattener options.
FlattenerPreset	String	The transparency flattener preset name.
FontOptions	PrintFontOptions object	The printing font options.
JobOptions	PrintJobOptions object	The printing job options.
PageMarksOptions	PrintPageMarksOptions object	The printing page marks options.
PaperOptions	PrintPaperOptions object	The paper options.
PostScriptOptions	PrintPostScriptOptions object	The PostScript options for printing.
PPDName	String	The PPD name.
PrinterName	String	The printer name.
PrintPreset	String	The print style.

► Setting print options

```
'Creates a new document and add some symbol items,
'creates a variety of print options and assign each
'to a PrintOptions object, then print the document
'with the combined PrintOptions object.
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
Set docRef = appRef.Documents.Add()
```

```
y = docRef.Height - 30
```

```
i = 1
```

```
Do While (i < 2)
```

```
    Set symbolRef = docRef.Symbols(i)
```

```
        Set itemRef = docRef.SymbolItems.Add(symbolRef)
        itemRef.Top = y
        itemRef.Left = 100
        y = (y - (itemRef.Height + 10))
        i = i + 1
    Loop
    appRef.Redraw

    'create multiple options and assign to PrintOptions object
    Set printOpts = CreateObject("Illustrator.printOptions")
    Set colorOpts = CreateObject("Illustrator.PrintColorManagementOptions")
    colorOpts.Name = "ColorMatch RGB"
    colorOpts.Intent = 1 'aiSaturationIntent
    printOpts.ColorManagementOptions = colorOpts

    Set jobOpts = CreateObject("Illustrator.PrintJobOptions")
    jobOpts.Designation = aiAllLayers
    jobOpts.ReversePages = True
    printOpts.JobOptions = jobOpts

    Set coordinateOpts = CreateObject("Illustrator.PrintCoordinateOptions")
    coordinateOpts.FitToPage = True
    printOpts.coordinateOptions = coordinateOpts

    Set flattenerOpts = CreateObject("Illustrator.PrintFlattenerOptions")
    flattenerOpts.ClipComplexRegions = True
    flattenerOpts.GradientResolution = 60
    flattenerOpts.RasterizationResolution = 60
    printOpts.FlattenerOptions = flattenerOpts

    docRef.PrintOut printerOpts
```

PrintPageMarksOptions

The options for printing page marks.

PrintPageMarksOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
BleedOffsetRect	Variant Array of 4 Doubles	The bleed offset rectangle.
ColorBars	Boolean	If true, enable color bars printing. Default: false
MarksOffsetRect	Variant Array of 4 Doubles	The page marks offset rectangle.
PageInfoMarks	Boolean	If true, enable page info marks printing. Default: false
PageMarksType	AiPageMarksStyle	The predefined page marks style name. Default: 1 ' aiPageMarksRoman
RegistrationMarks	Boolean	If true, enable registration marks printing. Default: false
TrimMarks	Boolean	If true, enable trim marks printing. Default: false
TrimMarksWeight	Double	Stroke weight of trim marks. Minimum: 0.0 Default: 0.125

► Setting page mark options

```
'Creates a new document with a rectangle, creates a
'PrintPageMarksOptions object and assigns it
'to a PrintOptions object sets some attributes then prints

'create a simple path item for printing
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add
Set pathItem = docRef.PathItems.Rectangle(600, 200, 200, 200)
docRef.GraphicStyles(2).ApplyTo pathItem

'Create a PrintCoordinateOptions object and assign it
'to a PrintOptions object
Set pageMarksOpts = CreateObject("Illustrator.PrintPageMarksOptions")
pageMarksOpts.ColorBars = True
pageMarksOpts.PageInfoMarks = True
pageMarksOpts.RegistrationMarks = True
pageMarksOpts.TrimMarks = True

Set printerOpts = CreateObject("Illustrator.printOptions")
printerOpts.PageMarksOptions = pageMarksOpts

'print it
docRef.PrintOut printerOpts
```

PrintPaperOptions

Contains information about the paper to be used in a print job.

PrintPaperOptions properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Height	Double	The custom height (in points) for using the custom paper. Default: 0 . 0
Name	String	The paper's name.
Offset	Double	Custom offset (in points) for using the custom paper. Default: 0 . 0
Transverse	Boolean	If <code>true</code> , transverse the artwork (rotate 90 degrees) on the custom paper. Default: <code>false</code>
Width	Double	The custom width in points, for using the custom paper. Default: 0 . 0

PrintPostScriptOptions

Specifies the options to be used when printing to a PostScript printer.

PrintPostScriptOptions properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
BinaryPrinting	Boolean	If <code>true</code> , print in binary mode. Default: <code>false</code>
CompatibleShading	Boolean	If <code>true</code> , use PostScript level 1 compatible gradient and gradient mesh printing. Default: <code>false</code>
ForceContinuousTone	Boolean	If <code>true</code> , force continuous tone. Default: <code>false</code>
ImageCompression	AiPostScriptImageCompressionType	The image compression type. Default: 0 ' <code>aiImageCompressionNone</code>
NegativePrinting	Boolean	If <code>true</code> , print in negative mode. Default: <code>false</code>
PostScriptLevel	AiPrinterPostScriptLevelEnum	The PostScript language level. Default: 2 ' <code>aiLevel2</code>
ShadingResolution	Double	The shading resolution Range: 1.0 to 9600.0. Default: 300.0

► Setting PostScript options

```
'Creates a new document, adds text, creates
'a new PrintPostScriptOptions object then
'prints with several postscript levels

'Create a new document and add a TextFrame
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()
Set textRef = docRef.TextFrames.Add()
textRef.Top = 600
textRef.Left = 50
textRef.Contents = "PrintPostScriptOptions object"

'Create a PrintPostScriptOptions object and
'assign it to a PrintOptions object.
Set psOpts = CreateObject("Illustrator.PrintPostScriptOptions")
Set printOpts = CreateObject("Illustrator.printOptions")
printOpts.PostScriptOptions = psOpts

'print with different PS levels
psOpts.PostScriptLevel = 2 'aiPSLevel2
docRef.PrintOut printOpts

psOpts.PostScriptLevel = 3 'aiPSLevel3
docRef.PrintOut printOpts
```

RasterItem

A bitmap art item in a document. A script can create raster items from an external file, and can create new raster items by copying and pasting or duplicating existing raster items.

RasterItem properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
ArtworkKnockout	AiKnockoutState	Is this object used to create a knockout, and if so, what kind.
BitsPerChannel	Integer	Read-only. The number of bits per channel.
BlendingMode	AiBlendModes	The mode used when compositing an object.
BoundingBox	Variant Array of 4 Doubles	Dimensions of item regardless of transformations.
Channels	Integer	Read-only. The number of channels.
Colorants	Array of Strings	Read-only. The colorants used in the raster art.
ColorizedGrayscale	Boolean	Read-only. If <code>true</code> , the raster art is a colorized grayscale image.
ContentVariable	Variable	The Variable bound to this item. It is not necessary to set the type before binding.
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Editable	Boolean	Read-only. If <code>true</code> , this item is editable.
Embedded	Boolean	If <code>true</code> , the Raster art can be embedded within the illustration.
File	String	Read-only. The file containing the art.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object excluding stroke width.
Height	Double	The height of the item, based on <code>GeometricBounds</code> .
Hidden	Boolean	If <code>true</code> , this item is hidden.
ImageColorSpace	AiImageColorSpace	Read-only. The color space of the Raster image.
IsIsolated	Boolean	If <code>true</code> , this object is isolated.
Layer	Layer object	Read-only. The layer this item belongs to.
Left	Double	The position of the left side of the item (in points, measured from the left side of the page).
Locked	Boolean	If <code>true</code> , this item is locked
Matrix	Matrix object	The transformation matrix applied to the item.

Property	Value type	What it is
Name	String	The name of this item.
Note	String	The note assigned to this object.
Opacity	Double	The opacity of the object. Range: 0.0 to 100.0
Overprint	Boolean	If <code>true</code> , the raster art overprints.
Parent	object	Read-only. The object that contains this item.
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
Selected	Boolean	If <code>true</code> , this item is selected.
Sliced	Boolean	If <code>true</code> , this item is sliced.
Status	AiRasterLinkState	Read-only. The status of the linked image, if the image is stored externally.
Tags	Tags collection object	Read-only. The tags contained in this item.
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).
Transparent	Boolean	Read-only. If <code>true</code> , the raster art is transparent.
URL	String	The value of the Adobe URL tag assigned to this item.
VisibilityVariable	Variable	The <code>Variable</code> bound to this item.
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the item including stroke width.
Width	Double	The width of the item, based on <code>GeometricBounds</code> .
WrapInside	Boolean	If <code>true</code> , the text frame object are wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If <code>true</code> , the text frame objects are wrapped around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this art item within the stacking order of the group or layer (<code>Parent</code>) that contains the art item.

RasterItem methods

Method	Returns	What it does
Colorize (rasterColor as object)	Nothing	Colorizes the RasterItem with a CMYK or RGB color.
Copy ()	Nothing	Copies the item to the clipboard; the associated document must be the frontmost document.
Cut ()	Nothing	Cuts the item to the clipboard; the associated document must be the frontmost document.
Delete ()	Nothing	Deletes the object.
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	Object	Duplicates the art item, optionally with the location and position for the copy.
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the art item to a new location and position.
Resize (scaleX as Double , scaleY as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the art item where scaleX is the horizontal scaling factor and scaleY is the vertical scaling factor; 100.0 = 100%
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the art item relative to the current rotation. The object is rotated counter-clockwise if the Angle value is positive, clockwise if the value is negative.
Trace ()	PluginItem object	Converts the raster art for this object to vector art, using default options. Reorders the raster art into the source art of a plugin group, and converts it into a group of filled and/or stroked paths that resemble the original image. Creates and returns a PluginItem object that references a TracingObject object.

Method	Returns	What it does
Transform (transformationMatrix as Matrix , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the art item by applying a transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the art item relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the art item's position in the stacking order of the group or layer (Parent) of this object.

RasterItems

A collection of `RasterItem` objects.

RasterItems properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. The number of elements in the collection.
Parent	Object	Read-only. The object's container.

RasterItems methods

Method	Returns	What it does
Index (item as <code>RasterItem</code>)	Long	Returns the index position of the object within the collection.
Item (itemKey)	<code>RasterItem</code> object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Creating a raster item

```
'Creates a new raster item in a new document from a file
'jpgFilePath contains the full path and file name of the file
```

```
Sub rasterItemCreate(jpgFilePath)
    Set app = CreateObject("Illustrator.Application")
    Set myDoc = app.Documents.Add()
    Set myPlacedItem = myDoc.PlacedItems.Add()
    myPlacedItem.File = jpgFilePath
    myPlacedItem.Embed()
End Sub
```

► Getting properties of raster items

```
'Extracts the color space type of the first raster item
```

```
Set appRef = CreateObject("Illustrator.Application")
Set rasterArt = appRef.ActiveDocument.RasterItems(1)
Select Case (rasterArt.ImageColorSpace)
    Case 3 'aiImageCMYK
        ' "The first raster item is a CMYK raster item"
    Case 2 'aiImageRGB
        ' "The first raster item is an RGB raster item"
    Case 1 'aiImageGrayscale
        ' "The first raster item is a Grayscale raster item"
End Select
```

RGBColor

An RGB color specification, used to apply an RGB color to a layer or art item.

If the color space of a document is RGB and you specify the color value for a page item in that document using CMYK, Illustrator will translate the CMYK color specification into an RGB color specification. The same thing happens if the document's color space is CMYK and you specify colors using RGB. Since this translation can lose information, you should specify colors using the class that matches the document's actual color space.

RGBColor properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Blue	Double	The blue color value. Range: 0.0 to 255.0
Green	Double	The green color value. Range: 0.0 to 255.0
Red	Double	The red color value. Range: 0.0 to 255.0

► Setting RGB colors

This script sets the default fill color of the frontmost document to yellow using an RGB object. If the color space of the frontmost document is CMYK, then Illustrator will regard the RGB fill color as a CMYK color although it is specified using RGB.

```
'Creates a new RGB color then applies the color to the first path item
```

```
Set appRef = CreateObject("Illustrator.Application")
Set newFillColor = CreateObject("Illustrator.RGBColor")
Set frontPath = appRef.ActiveDocument.PathItems(1)
```

```
'Define the new color
newFillColor.Red = 255
newFillColor.Green = 255
newFillColor.Blue = 0
```

```
frontPath.Filled = True
frontPath.fillColor = newFillColor
```

Screen

Associates a color separation screen with information to be used for printing.

Screen properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Name	String	The color separation screen.
ScreenInfo	ScreenInfo object	The color separation screen information.

ScreenInfo

Contains information about the angle and frequency of the screen to be used for printing.

ScreenInfo properties

Property	Value type	What it is
Angle	Double	The screen's angle in degrees.
Application	Application object	Read-only. The Illustrator Application object.
DefaultScreen	Boolean	If true, it is the default screen.
Frequency	Double	The screen's frequency.

► Getting screen information

```
'Creates a document and displays screen data available
'for first PPD file, in a text frame

'Create a new document and add a TextFrame
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()
Set textRef = docRef.TextFrames.Add()
textRef.Top = 600
textRef.Left = 50
textRef.Contents = "Screen Objects for 1st PPD File: " & vbCrLf

'Get the first PPD
ppdFileList = appRef.PPDFileList
If IsEmpty(ppdFileList) Then
    sInfo = "Empty PPDFileList"
Else
    Set ppdFile = ppdFileList(0)

    sInfo = ppdFile.Name & vbCrLf
    If IsEmpty(screenList) Then
        sInfo = sInfo & "Empty ScreenList" & vbCrLf
```



```
Else
    'Get info on the all screens from the 1st PPD
    For Each screenRef In ppdFile.PPDInfo.ScreenList
        sInfo = sInfo & screenRef.Name & vbCrLf
        sInfo = sInfo & vbTab & "Angle = "
        sInfo = sInfo & CStr(screenRef.ScreenInfo.Angle) & vbCrLf
        sInfo = sInfo & vbTab & "Frequency = "
        sInfo = sInfo & screenRef.ScreenInfo.Frequency & vbCrLf
    Next
End If
End If
textRef.Contents = textRef.Contents & sInfo
appRef.Redraw
```

ScreenSpotFunction

Contains information about the screen spot function including its definition in PostScript language code.

ScreenSpotFunction properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Name	String	The color separation screen spot function name.
SpotFunction	String	The spot function expressed in PostScript language commands.

► Getting screen spot information

```
'Creates a new document and displays ScreenSpot
'data available for first PPD file, in a text frame

'Create a new document and add a TextFrame
Set appRef = CreateObject("Illustrator.Application")

Set docRef = appRef.Documents.Add()
Set textRef = docRef.TextFrames.Add()
textRef.Top = 600
textRef.Left = 50
textRef.Contents = "ScreenSpotFunctions for 1st PPD:" & vbCrLf

'Get the first PPD
ppdFileList = appRef.PPDFileList
If IsEmpty(ppdFileList) Then
    sInfo = "Empty PPDFileList"
Else
    Set ppdRef = ppdFileList(0)

    sInfo = ppdRef.Name & vbCrLf
    If IsEmpty(screenSpotFunctionList) Then
        sInfo = sInfo & "Empty ScreenSpotFunctionList" & vbCrLf
    Else
        'Get data on the all ScreenSpots from the 1st PPD
        For Each screenSpot In ppdRef.PPDInfo.ScreenSpotFunctionList
            sInfo = sInfo + screenSpot.Name & ": "
            sInfo = sInfo + screenSpot.SpotFunction & vbCrLf
            sInfo = sInfo & vbCrLf
        Next
    End If
End If
textRef.Contents = textRef.Contents + sInfo
appRef.Redraw
```

Spot

A custom color definition contained in a `SpotColor` object.

If no properties are specified when creating a new spot, default values are provided. However, if specifying the color, you must use the same color space as the document, either CMYK or RGB. Otherwise, an error results. When created, the spot is added to the end of the swatches list in the Swatches palette.

Spot properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Color	Color object	The color information for this spot color.
ColorType	AiColorModel	The color model for this spot color.
Name	String	The spot color's name.
Parent	Document object	Read-only. The document that contains this spot color.

Spot methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.

► Creating a new spot

'Creates and adds a new spot color to the current document

```
Set appRef = CreateObject("Illustrator.Application")
```

'Create the new spot

```
Set frontDocument = appRef.ActiveDocument
```

```
Set newSpot = frontDocument.Spots.Add
```

'Define the new color value

```
Set newColor = CreateObject("Illustrator.CMYKColor")
```

```
newColor.Cyan = 35
```

```
newColor.Magenta = 0
```

```
newColor.Yellow = 50
```

```
newColor.Black = 0
```

'Define a new `SpotColor` with an 80% tint

'of the new Spot's color. The spot color can then

'be applied to an art item like any other color.

```
newSpot.Name = "Pea-Green"
```

```
newSpot.ColorType = 2 'aiSpot
```

```
newSpot.Color = newColor
```

```
Set newSpotColor = CreateObject("Illustrator.SpotColor")
```

```
newSpotColor.Spot = newSpot
```

SpotColor

Color class used to apply the color value of a spot at a specified tint value.

SpotColor properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Spot	Spot object	A reference to the Spot object that defines the color.
Tint	Double	The tint of the color. Range: 0.0 to 100.0

► Applying a tint to a spot

Your frontmost document must already have a PathItem before you can run this script.

```
'Creates a new RGB color, creates a new spot color,
'sets the color of the spot to 80% of the new RGB color,
'then applies 50% of the spot color to the frontmost path item

Set appRef = CreateObject("Illustrator.Application")
Set newColor = CreateObject("Illustrator.RGBColor")

'Define the new color value
newColor.Red = 255
newColor.Green = 0
newColor.Blue = 0

'Create the new spot
Set frontDocument = appRef.ActiveDocument
Set newSpot = frontDocument.Spots.Add

'Define the new SpotColor as 80% of the specified RGB color
newSpot.Name = "Red spot color"
newSpot.Color = newColor

'Apply 50% of the spot color just created to the frontmost path item,
'by creating a SpotColor object, setting its properties,
'then applying that to a path item's fill color.
Set newSpotColor = CreateObject("Illustrator.SpotColor")
newSpotColor.Spot = newSpot
newSpotColor.Tint = 50

Set frontPath = frontDocument.PathItems(1)
frontPath.Filled = True
frontPath.FillColor = newSpotColor
```

Spots

A collection of spot colors in a document.

Spots properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Document	Read-only. The object's document container.

Spots methods

Method	Returns	What it does
Add ()	Spot object	Creates a new <code>Spot</code> object.
Index (item as <code>Spot</code>)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Spot object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Removing spots from a document

This script illustrates how to remove all spots defined in the frontmost document.

```
'Deletes all spots from the current document

Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.ActiveDocument
spotCount = docRef.Spots.Count
If (spotCount > 0) Then
    docRef.Spots.RemoveAll
End If
```

Story

A contiguous block of text as specified by a text range. A story can contain one or more text frames; if there is more than one, the multiple text frames are linked together to form a single story.

Story properties

Property	Value Type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Characters	Characters	Read-only. All of the characters in the story.
InsertionPoints	InsertionPoints	Read-only. All of the insertion points in this story.
Length	Long	Read-only. The number of characters in the story.
Lines	Lines	Read-only. All of the lines in this story.
Paragraphs	Paragraphs	Read-only. All of the paragraphs in this story.
Parent	Object	Read-only. The object's container.
TextFrames	TextFrames	Read-only. The text frame items in this story.
TextRange	TextRange	Read-only. The text range of the story.
TextRanges	TextRanges	Read-only. All of the text ranges in the story.
TextSelection	Variant Array of TextRange objects	Read-only. The selected text ranges in the story.
Words	Words	Read-only. All of the words in the story.

► Linking text frames into a story

```
'Creates a new document with 3 text frames, creates a
'story flowing through the first 2 text frames then
'creates another story fully contained in the 3rd text frame
```

```
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()
```

```
'Create the first text frame
Set itemRef1 = docRef.PathItems.Rectangle(600, 200, 50, 30)
Set textRef1 = docRef.TextFrames.AreaText(itemRef1)
textRef1.Selected = True
```

```
'create the 2nd text frame and
'link it to the first
Set itemRef2 = docRef.PathItems.Rectangle(550, 300, 50, 200)
Set textRef2 = docRef.TextFrames.AreaText(itemRef2, aiHorizontal, textRef1)
textRef1.Contents = "This is two text frames linked together as one story"
textRef2.Selected = True
appRef.Redraw
```

```
'Create a 3rd text frame and count the stories
Set textRef3 = docRef.TextFrames.Add
textRef3.Contents = "Each unlinked textFrame adds a story."
textRef3.Top = 650
textRef3.Left = 200
appRef.Redraw
```

Stories

A collection of `Story` objects. See [Story](#) for an example.

Stories properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Document	Read-only. The object's document container.

Stories methods

Method	Returns	What it does
Index (item as Story)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Story object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).

Swatch

A color swatch definition contained in a document. The swatches correspond to the swatch palette in the Illustrator user interface. Additional swatches can be created either manually by a user or by a script. The swatch can hold all types of color data (that is, pattern, gradient, CMYK, RGB, gray, and spot).

Swatch properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Color	Color object	The color information for this swatch.
Name	String	The swatch's name.
Parent	Document object	Read-only. The document that contains this swatch.

Swatch methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.

► Changing the name of a swatch

This script illustrates how to change the name of the fifth swatch.

```
'Changes the name of a swatch
```

```
Set appRef = CreateObject("Illustrator.Application")
```

```
Set swatch5 = appRef.ActiveDocument.Swatches(5)
```

```
swatch5.Name = "myColor"
```

Swatches

A collection of `swatch` objects in a document.

Swatches properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Document	Read-only. The object's document container.

Swatches methods

Method	Returns	What it does
Add ()	Swatch object	Creates a new <code>Swatch</code> object.
Index (item as <code>Swatch</code>)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Swatch object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Deleting a swatch

This script illustrates how to first obtain a swatch by index and then how to delete that swatch.

```
'Deletes a swatch from the current document

Set appRef = CreateObject("Illustrator.Application")

Set documentSwatches = appRef.ActiveDocument.Swatches
Set swatchToDelete = documentSwatches(5)
swatchToDelete.delete
```

Symbol

An art item that is stored in the Symbols palette, and can be reused one or more times in the document without duplicating the art data. Symbols are contained in documents. Instances of `Symbol` in a document are associated with `SymbolItem` objects, which store the art object properties.

Symbol properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Name	String	The symbol's name.
Parent	Document object	Read-only. The document that contains this symbol.

Symbol methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.
Duplicate ()	object	Creates a duplicate of the object.

Symbols

A collection of `symbol` objects in a document.

Symbols properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Document	Read-only. The object's document container.

Symbols methods

Method	Returns	What it does
Add (sourceArt as Object)	Symbol object	Creates a new <code>Symbol</code> object in the document based on the art item.
Index (item as Symbol)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Symbol object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Creating symbols

```
'Creates a path item to display each graphic style,
'then adds each item as a new symbol

Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()
y = docRef.Height - 30

iCount = docRef.GraphicStyles.Count

i = 1
Do While (i < (iCount + 1))
    'create a PathItem and apply the style to it
    Set styleRef = docRef.GraphicStyles(i)
    Set itemRef = docRef.PathItems.Rectangle(y, 100, 20, 20)
    styleRef.ApplyTo itemRef

    'create a new symbol from the graphic style
    docRef.Symbols.Add itemRef

    y = (y - (itemRef.Height + 40)) ' reduce y for next item
    i = i + 1
Loop

appRef.Redraw
```

SymbolItem

An art item made reusable by adding it to the Symbols palette. A `SymbolItem` is linked to the `Symbol` from which it was created and changes if you modify the associated `Symbol` object.

SymbolItem properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
ArtworkKnockout	AiKnockoutState	Is this object used to create a knockout, and if so, what kind.
BlendingMode	AiBlendModes	The mode used when compositing an object.
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Editable	Boolean	Read-only. If <code>true</code> , this item is editable.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object excluding stroke width.
Height	Double	The height of the item, based on <code>GeometricBounds</code> .
Hidden	Boolean	If <code>true</code> , this item is hidden.
IsIsolated	Boolean	If <code>true</code> , this object is isolated.
Layer	Layer object	Read-only. The layer this item belongs to.
Left	Double	The position of the left side of the item (in points, measured from the left side of the page).
Locked	Boolean	If <code>true</code> , this item is locked
Name	String	The name of this item.
Note	String	The note assigned to this item.
Opacity	Double	The opacity of the object. Range: 0.0 to 100.0
Parent	object	Read-only. The object that contains this item.
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
Selected	Boolean	If <code>true</code> , this item is selected.
Sliced	Boolean	If <code>true</code> , this item is sliced.
Symbol	Symbol object	The symbol object to which this item is linked. You can set this property to link it to a different symbol object.
Tags	Tags collection object	Read-only. The tags contained in this item.

Property	Value type	What it is
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).
URL	String	The value of the Adobe URL tag assigned to this item.
VisibilityVariable	Variable	The Variable bound to this item.
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the item including stroke width.
Width	Double	The width of the item, based on <code>GeometricBounds</code> .
WrapInside	Boolean	If <code>true</code> , the text frame object are wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If <code>true</code> , the text frame objects are wrapped around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this art item within the stacking order of the group or layer (<code>Parent</code>) that contains the art item.

SymbolItem methods

Method	Returns	What it does
Copy ()	Nothing	Copies the item to the clipboard; the associated document must be the frontmost document.
Cut ()	Nothing	Cuts the item to the clipboard; the associated document must be the frontmost document.
Delete ()	Nothing	Deletes the object.
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	Object	Duplicates the art item, optionally with the location and position for the copy.
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the art item to a new location and position.

Method	Returns	What it does
Resize (scaleX as Double , scaleY as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the art item where <code>scaleX</code> is the horizontal scaling factor and <code>scaleY</code> is the vertical scaling factor; 100.0 = 100%
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the art item relative to the current rotation. The object is rotated counter-clockwise if the <code>Angle</code> value is positive, clockwise if the value is negative.
Transform (transformationMatrix as Matrix , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the art item by applying a transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the art item relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the art item's position in the stacking order of the group or layer (Parent) of this object.

► Getting symbol items for symbols

'Creates a document and adds every symbol item to it

```
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()
```

```
y = docRef.Height - 30
x = 50
i = 1
```

```
Do While (i < (docRef.Symbols.Count + 1))
    Set symbolRef = docRef.Symbols(i)
    Set itemRef = docRef.SymbolItems.Add(symbolRef)
    itemRef.Top = y
    itemRef.Left = x
    y = y - (itemRef.Height + 20)
    If (y < 60) Then
```

```
        y = docRef.Height - 30
        x = x + 200
    End If
    i = i + 1
Loop
```


SymbolItems

A collection of `SymbolItem` objects in a document. See [SymbolItem](#) for example.

SymbolItems properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	object	Read-only. The object's container.

SymbolItems methods

Method	Returns	What it does
Add (symbol as Symbol)	<code>SymbolItem</code> object	Creates a new <code>SymbolItem</code> object in the document linked to the given symbol.
Index (item as SymbolItem)	Long	Returns the index position of the object within the collection.
Item (itemKey)	<code>SymbolItem</code> object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

TabStopInfo

Information about the alignment, position, and other details for a tab stop in a ParagraphAttributes object.

TabStopInfo properties

Property	Value type	What it is
Alignment	AiTabStopAlignment	The alignment of the tab stop. Default: 0 ' aiLeftTab
Application	Application object	Read-only. The Illustrator Application object.
DecimalCharacter	String	The character used for decimal tab stops. Default: .
Leader	String	The leader dot.
Position	Double	The position of the tab stop expressed in points. Default: 0.0

► Getting tab stop information

```
'Iterates through each text frame and display
'data found about each tab stop, if any

'verify a document is open
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.ActiveDocument
For Each textRef In docRef.TextFrames
    ' check each TextFrame for paragraphs
    For Each paraRef In textRef.Paragraphs
        ' check each paragraph for TabStops
        For Each tabRef In paraRef.ParagraphAttributes.TabStops
            sData = "TabStop Leader = " & tabRef.Leader & Chr(9)
            sData = sData & "TabStop Position = " & CStr(tabRef.Position) &
Chr(10)
        Next
    Next
Next

Set newTF = docRef.TextFrames.Add
newTF.Contents = sData
newTF.Top = 400
newTF.Left = 100
```

Tag

A tag associated with a specific art item. Tags allow you to assign an unlimited number of key-value pairs to any item in a document.

Tag properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Name	String	The Tag's name.
Parent	Object	Read-only. The object that contains this Tag.
Value	String	The data stored in this Tag.

Tag methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.

► Using tags

```
'Displays tag information of selected item in a text frame
'in a new document

Set appRef = CreateObject("Illustrator.Application")
newTagName = "OneWord"
selection = appRef.selection
If (Not IsEmpty(selection)) Then
    'Get the first selected item
    Set selectedArt = selection(0)
    Set tagList = selectedArt.Tags
    If tagList.Count = 0 Then
        Set tempTag = tagList.add
        tempTag.Name = newTagName
        tempTag.Value = "anything you want"
    End If

    'Create a document and add a line of text per tag
    Set reportDocument = appRef.Documents.Add
    top_offset = 400
    For Each tagItem In tagList
        Set tagText = reportDocument.TextFrames.Add
        tagText.Contents = "Tag " & Chr(13) & Chr(9) & tagItem.Name & Chr(13)
        & Chr(9) & tagItem.Value & Chr(10)
        tagText.Position = Array(100, top_offset)
        top_offset = top_offset - 20
    Next
End If
```

Tags

A collection of `Tag` objects.

Tags properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	object	Read-only. The object's container.

Tags methods

Method	Returns	What it does
Add ()	Tag object	Creates a new <code>Tag</code> object.
Index (item as Tag)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Tag object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Creating tags to mark images

This example illustrates how to mark all images in a document with a specific tag. If your script creates temporary `PageItems`, you can then later look at the "MyInfo" tag. If it exist for a particular `PageItem` and it's value is "OriginalItem" you know not to delete it.

```
'Creates name and value tags for each placed or raster
'item in the current document

Set appRef = CreateObject("Illustrator.Application")

Set frontDocument = appRef.ActiveDocument
For Each imageArt In frontDocument.PageItems
    If ((imageArt.PageItemType = 6) Or _
        (imageArt.PageItemType = 8)) Then

        'Create a new Tag with the name myInfo and the value originalItem
        Set tagRef = imageArt.Tags.Add
        tagRef.Name = "MyInfo"
        tagRef.Value = "OriginalItem"
    End If
Next
```

TextFont

Information about a font in the document, found in a [CharacterAttributes](#) object.

TextFont properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator Application object.
Family	String	Read-only. The font's family name.
Name	String	Read-only. The font's full name.
Parent	Object	Read-only. The object's container.
Style	String	Read-only. The font's style name.

► Showing fonts in a document

```
'Creates a new A3 sized document and lists available
'fonts until the document is full.

Set appRef = CreateObject("Illustrator.Application")
myWidth = 1191.0
myHeight = 842.0
Set docRef = appRef.Documents.Add(, myWidth, myHeight)
edgeSpacing = 10
columnSpacing = 230

x = edgeSpacing
y = docRef.Height - edgeSpacing

iCounter = 0
For Each fontRef In appRef.TextFonts
    'create the text frame
    Set textRef = docRef.TextFrames.Add()
    textRef.TextRange.CharacterAttributes.Size = 10
    textRef.Contents = fontRef.Name + " " + fontRef.Style
    textRef.Top = y
    textRef.Left = x

    ' Check whether the new text frame has gone over the document bounds
    if((x + textRef.Width) > docRef.Width) Then
        ' delete the text frame
        textRef.Delete()
        Exit For
    Else
        ' keep the text frame
        appRef.Redraw
        textRef.TextRange.CharacterAttributes.TextFont =
appRef.TextFonts.Item(fontRef.Name)
        y = y - textRef.Height
        If (y < 20) Then'reset y,x at bottom of page
            y = docRef.Height - edgeSpacing
```

```
        x = x + columnSpacing
    End If
End If
iCounter = iCounter + 1
Next
```

TextFonts

A collection of `TextFont` objects.

TextFonts properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	object	Read-only. The object's container.

TextFonts methods

Method	Returns	What it does
Index (item as <code>TextFont</code>)	Long	Returns the index position of the object within the collection.
Item (itemKey)	<code>TextFont</code> object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).

► Finding a font

```
'Searches through the list of available fonts for the "Symbol" font

Set appRef = CreateObject("Illustrator.Application")

fontName = "Symbol"
foundTextFace = False
For Each fontToTest In appRef.TextFonts
    If (fontToTest.Name = fontName) Then
        foundTextFace = True
    End If
Next

If (foundTextFace) Then
    'fontName & " is installed on this machine"
Else
    'fontName & " is not installed on this machine"
End If
```

TextFrame

The basic art item for displaying text. From the user interface, this is text created with the Text tool. There are three types of text art in Illustrator: point text, path text, and area text. The type is indicated by the text frame's [Kind](#) property.

When you create a text frame, you also create a [Story](#) object. However, threading text frames combines the frames into a single story object. To thread frames, use the [NextFrame](#) or [PreviousFrame](#) property.

TextFrame properties

Property	Value type	What it is
Anchor	VariantArray of 2 Doubles	The position of the anchor point along the path text.
Application	Application object	Read-only. The Illustrator Application object.
ArtworkKnockout	AiKnockoutState	Is this object used to create a knockout, and if so, what kind.
BlendingMode	AiBlendModes	The mode used when compositing an object.
Characters	Characters	Read-only. All the characters in this text frame.
ColumnCount	Long	The column count in the text frame (area text only).
ColumnGutter	Double	The column gutter in the text frame (area text only).
Contents	String	The text contents of this text frame.
ContentVariable	Variable	The content variable bound to this text frame.
ControlBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object including stroke width and controls.
Editable	Boolean	Read-only. If <code>true</code> , this item is editable.
EndTValue	Double	The end position of text along a path, as a value relative to the path's segments (path text only).
FlowsLinkHorizontally	Boolean	If <code>true</code> , text is first flowed horizontally between linked frames.
GeometricBounds	Variant Array of 4 Doubles	Read-only. The bounds of the object excluding stroke width.
Height	Double	The height of the item, based on <code>GeometricBounds</code> .
Hidden	Boolean	If <code>true</code> , this item is hidden.
InsertionPoints	InsertionPoints	Read-only. All the insertion points in this text frame.
IsIsolated	Boolean	If <code>true</code> , this object is isolated.
Kind	AiTextType	Read-only. The type of text frame item.

Property	Value type	What it is
Layer	Layer object	Read-only. The layer this item belongs to.
Left	Double	The position of the left side of the item (in points, measured from the left side of the page).
Lines	Lines	Read-only. All the lines in this text frame.
Locked	Boolean	If <code>true</code> , this item is locked.
Matrix	Matrix	Read-only. The transformation matrix for this text frame.
Name	String	The name of this item.
Note	String	The note assigned to this item.
NextFrame	TextFrame	The linked frame following this one.
Opacity	Double	The opacity of the object. Range: 0.0 to 100.0
OpticalAlignment	Boolean	If <code>true</code> , the optical alignment is active.
Orientation	AiTextOrientation	The orientation of the text in the frame.
Paragraphs	Paragraphs	Read-only. All the paragraphs in this text frame.
Parent	object	Read-only. The object that contains this item.
Position	Variant Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
PreviousFrame	TextFrame	The linked text frame preceding this one.
RowCount	Long	The row count in the text frame (area text only).
RowGutter	Double	The row gutter in the text frame (area text only).
Selected	Boolean	If <code>true</code> , this item is selected.
Sliced	Boolean	If <code>true</code> , this item is sliced.
Spacing	Double	The amount of spacing between characters that flow around a sharp curve or acute angle in the path. Note: Valid only when Kind is path.
StartTValue	Double	The start position of text along a path, as a value relative to the path's segments. Note: Valid only when Kind is path.
Story	Story	Read-only. The story to which the text frame belongs.
Tags	Tags collection object	Read-only. The tags contained in this item.
TextPath	TextPath	Read-only. The path item associated with the text frame. Note: Valid only when Kind is area or path.

Property	Value type	What it is
TextRange	TextRange	Read-only. The text range of the text frame.
TextRanges	TextRanges	Read-only. All the text in this text frame.
TextSelection	Variant Array of TextRange	Read-only. The selected text (ranges) in the TextFrame.
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).
URL	String	The value of the Adobe URL tag assigned to this item.
VisibilityVariable	Variable	The Variable bound to this item.
VisibleBounds	Variant Array of 4 Doubles	Read-only. The visible bounds of the item including stroke width.
Width	Double	The width of the item, based on GeometricBounds.
Words	Words	Read-only. All the words in this text frame.
WrapInside	Boolean	If true, the text frame object are wrapped inside this object.
WrapOffset	Double	The offset to use when wrapping text around this object.
Wrapped	Boolean	If true, the text frame objects are wrapped around this object (text frame must be above the object).
ZOrderPosition	Long	Read-only. The position of this art item within the stacking order of the group or layer (Parent) that contains the art item.

TextFrame methods

Method	Returns	What it does
CreateOutline ()	GroupItem object	Converts the frame's text to outlines.
Duplicate ([relativeObject as Object] [, insertionLocation as AiElementPlacement])	Object	Duplicates the art item, optionally specifying the location and position for the copy.
Move (relativeObject as Object, insertionLocation as AiElementPlacement)	Nothing	Moves the art item, specifying the new location and position.
Remove ()	Nothing	Deletes this object.

Resize (scaleX as Double , scaleY as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, scaleAbout as AiTransformation])	Nothing	Scales the art item where <code>scaleX</code> is the horizontal scaling factor and <code>scaleY</code> is the vertical scaling factor; 100.0 = 100%
Rotate (angle as Double , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, rotateAbout as AiTransformation])	Nothing	Rotates the art item relative to the current rotation. The object is rotated counter-clockwise if the <code>Angle</code> value is positive, clockwise if the value is negative.
Transform (transformationMatrix as Matrix , [, changePositions as Boolean] [, changeFillPatterns as Boolean] [, changeFillGradients as Boolean] [, changeStrokePattern as Boolean] [, changeLineWidths as Double] [, transformAbout as AiTransformation])	Nothing	Transforms the art item by applying a transformation matrix.
Translate ([deltaX as Double] [, deltaY as Double] [, transformObjects as Boolean] [, transformFillPatterns as Boolean] [, transformFillGradients as Boolean] [, transformStrokePatterns as Boolean])	Nothing	Repositions the art item relative to the current position, where <code>deltaX</code> is the horizontal offset and <code>deltaY</code> is the vertical offset.
ZOrder (zOrderCmd as AiZOrderMethod)	Nothing	Arranges the art item's position in the stacking order of the group or layer (Parent) of this object.

► Using text frames

```

'Duplicates and rotates the selected text art item 5 times

Set appRef = CreateObject("Illustrator.Application")

'First check the selection of the application
'It has to be a text art item in order for this script to run
selection = appRef.selection
Set frontDocument = appRef.ActiveDocument
Set sourceTextArt = selection(0)

'Get the parent of the text art so new text art items can be inserted
'in the same group or layer as the selected text art is in
Set textArtGroup = sourceTextArt.Parent.TextFrames

'Create 5 new versions of the text art each rotated a bit
For i = 1 To 5
    Set newTextArt = textArtGroup.Add
    newTextArt.Position = sourceTextArt.Position

```

```
newTextArt.Contents = sourceTextArt.Contents
newTextArt.Rotate 180 * i / 6
Next
```

TextFrames

A collection of `TextFrame` objects.

TextFrames properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	object	Read-only. The object's container.

TextFrames methods

Method	Returns	What it does
Add ()	<code>TextFrame</code> object	Creates a new point <code>TextFrame</code> object.
AreaText (textPath as <code>PathItem</code> [, orientation as AiTextOrientation] [, baseFrame as <code>TextFrame</code>] [, postfix as <code>Boolean</code>])	<code>TextFrame</code> object	Creates a new area <code>TextFrame</code> object.
Index (item as <code>TextFrame</code>)	Long	Returns the index position of the object within the collection.
Item (itemKey)	<code>TextFrame</code> object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
PathText (textPath as <code>PathItem</code> [, startTValue as <code>Double</code>] [, endTValue as <code>Double</code>] [, orientation as AiTextOrientation] [, baseFrame as <code>TextFrame</code>] [, postfix as <code>Boolean</code>])	<code>TextFrame</code> object	Creates an on-path text frame item.
PointText (anchor as <code>Array of 2 Doubles</code> [, orientation as AiTextOrientation])	<code>TextFrame</code> object	Creates a point text frame item.
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Adding 3 types of text frames

```
'Creates a document with 3 TextFrames (area, path, point),
'changes the contents of each TextFrame then deletes a text frame
```

```
Set appRef = CreateObject("Illustrator.Application")
Set docRef = appRef.Documents.Add()
```

```
'create 3 new textFrames (are, line, point)
'Area Text
Set rectRef = docRef.PathItems.Rectangle(700, 50, 100, 100)
Set areaTextRef = docRef.TextFrames.AreaText(rectRef)
areaTextRef.Contents = "TextFrame #1"
areaTextRef.Selected = True

'Line Text
Set lineRef = docRef.PathItems.Add()
lineRef.SetEntirePath (Array(Array(200, 700), Array(300, 550)))
Set pathTextRef = docRef.TextFrames.PathText(lineRef)
pathTextRef.Contents = "TextFrame #2"
pathTextRef.Selected = True

'Point Text
Set pointTextRef = docRef.TextFrames.Add()
pointTextRef.Contents = "TextFrame #3"
pointTextRef.Top = 700
pointTextRef.Left = 400
pointTextRef.Selected = True
appRef.Redraw

sText = "There are " & CStr(docRef.TextFrames.Count) & " TextFrames."
sText = sText & vbCrLf & "Changing contents of each TextFrame."

'change the content of each
areaTextRef.Contents = "Area TextFrame."
pathTextRef.Contents = "Path TextFrame."
pointTextRef.Contents = "Point TextFrame."
appRef.Redraw

docRef.TextFrames(2).Delete
appRef.Redraw
```

TextPath

A path or list of paths for area or path text. A path consists of path points that define its geometry.

TextPath properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
Area	Double	Read-only. The area of this path in square points. If the area is negative, the path is wound counterclockwise. Self-intersecting paths can contain sub-areas that cancel each other out, which makes this value zero even though the path has apparent area.
BlendingMode	AiBlendModes	The blend mode used when compositing an object.
Clipping	Boolean	If <code>true</code> , this path are used as a clipping path.
Editable	Boolean	Read-only. If <code>true</code> , this item is editable.
Evenodd	Boolean	If <code>true</code> , the even-odd rule are used to determine insideness.
FillColor	Color Object	The fill color of the path.
Filled	Boolean	If <code>true</code> , the path be filled.
FillOverprint	Boolean	If <code>true</code> , the art beneath a filled object are overprinted.
Guides	Boolean	If <code>true</code> , this path is a guide object.
Height	Double	The height of the group item.
Left	Double	The position of the left side of the item (in points, measured from the left side of the page).
Note	String	The note text assigned to the path.
Opacity	Double	The opacity of the object. Range: 0.0 to 100.0
Parent	Layer or GroupItem	Read-only. The parent of this object.
PathPoints	PathPoints	Read-only. The path points contained in this path item.
Polarity	AiPolarityValues	The polarity of the path.
Position	Array of 2 Doubles	The position (in points) of the top left corner of the item in the format [x, y]. Does not include stroke weight.
Resolution	Double	The resolution of the path in dots per inch (dpi).
SelectedPathPoints	PathPoints	Read-only. All of the selected path points in the path.
StrokeCap	AiStrokeCap	The type of line capping.
StrokeColor	Color Object	The stroke color for the path.
Stroke	Boolean	If <code>true</code> , the path are stroked.

Property	Value type	What it is
StrokeDashes	Object	The lengths for dashes and gaps in dashed lines, starting with the first dash length, followed by the first gap length, and so on. Set to an empty list, [], for a solid line.
StrokeDashOffset	Double	The default distance into the dash pattern at which the pattern are started.
StrokeJoin	AiStrokeJoin	Type of joints for the path.
StrokeMiterLimit	Double	When a default stroke join is set to <code>mitered</code> , this property specifies when the join will be converted to beveled (squared-off) by default. The default miter limit of 4 means that when the length of the point reaches four times the stroke weight, the join switches from a miter join to a bevel join. A value of 1 specifies a bevel join. Range: 1 to 500 Default: 4
StrokeOverprint	Boolean	If <code>true</code> , the art beneath a stroked object are overprinted.
StrokeWidth	Double	Width of the stroke.
Top	Double	The position of the top of the item (in points, measured from the bottom of the page).
Width	Double	The width of the item.

TextPath methods

Method	Returns	What it does
SetEntirePath (pathPoints as Array of [x, y] coordinate pairs)	Nothing	Sets the path using the array of points specified as [x, y] coordinate pairs.

TextRange

A range of characters from a text item (story, text frame, character, word, line, paragraph, or another text range).

TextRange properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator Application object.
CharacterAttributes	CharacterAttributes	Read-only. The character properties for the text range.
CharacterOffset	Long	Offset of the first character.
Characters	Characters	Read-only. All of the characters in the text range.
CharacterStyles	CharacterStyles	Read-only. List of referenced character styles in the text range.
Contents	String	The text string.
InsertionPoints	InsertionPoints	Read-only. All of the insertion points in this text range.
Kerning	Long	The spacing between two characters, in thousandths of an em.
Length	Long	The length (in characters). Minimum: 0
Lines	Lines	Read-only. All the lines in this text range.
ParagraphAttributes	ParagraphAttributes	Read-only. The paragraph properties.
Paragraphs	Paragraphs	Read-only. All of the paragraphs in this text range.
ParagraphStyles	ParagraphStyles	Read-only. A list of referenced paragraph styles in the text range.
Parent	Object	Read-only. The object's container.
Story	Story	Read-only. The story of the text range.
TextRanges	TextRanges	Read-only. All of the text in this text range.
TextSelection	Array of TextRange objects	Read-only. The selected text (ranges) in the text range.
Words	Words	Read-only. All of the words in this text range.

TextRange methods

Method	Returns	What it does
ChangeCaseTo (type as AiCaseChangeType)	Nothing	Changes the capitalization of text.
Delete ()	Nothing	Deletes the object.
DeSelect ()	Nothing	Deselects the text range.
Duplicate ([relativeObject as Object], [insertionLocation as AiElementPlacement])	TextRange	Duplicates the art item, optionally specifying the location and position for the copy.
Move (relativeObject as Object , insertionLocation as AiElementPlacement)	Nothing	Moves the art item to the new location and position.
Select ([addToDocument as Boolean])	Nothing	Selects the text range. If <code>addToDocument</code> is <code>true</code> , adds this to the current selection; otherwise replaces the current selection.

► Using a text range

```
'Increases the font size of the first 2 characters of each
'word in the current document

Set appRef = CreateObject("Illustrator.Application")

For Each textArt In appRef.ActiveDocument.TextFrames
    Set textArtRange = textArt.TextRange

    For Each textWord In textArtRange.Words
        'If word is longer than 2 characters
        'resize just the first 2 characters
        'otherwise, resize the whole word
        wordLen = Len(textWord.Contents)
        If (wordLen < 2) Then
            charsToChange = wordLen
        Else
            charsToChange = 2
        End If
        'resize the word
        If (charsToChange > 0) Then
            'Omit 1st arg to go from the beginning to
            'character number charsToChange-1 (First character is index 0)
            Dim i
            For i = 1 To charsToChange
                Set firstChars = textWord.Characters(i)
                firstChars.CharacterAttributes.Size =_
                    firstChars.CharacterAttributes.Size * 1.5
            Next
```

```
        End If
    Next
Next
```

TextRanges

A collection of `TextRange` objects.

TextRanges properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Document	Read-only. The object's document container.

TextRanges methods

Method	Returns	What it does
Index (item as TextRange)	Long	Returns the index position of the object within the collection.
Item (itemKey)	<code>TextRange</code> object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

TracingObject

A tracing object, which associates a source raster art item with a vector art plug-in group created by tracing. Scripts can initiate tracing using `PlacedItem.Trace` or `RasterItem.Trace`. The resulting `PluginItem` object represents the vector art group, and has this object in its `Tracing` property.

A script can force the tracing operation by calling the application's `Redraw` method. The operation is asynchronous, so a script should call `Redraw` after creating the tracing object, but before accessing its properties or expanding the tracing to convert it to an art item group.

The read-only properties that describe the tracing result have valid values only after the first tracing operation completes. A value of 0 indicates that the operation has not yet been completed.

TracingObject properties

Property	Value type	What it is
AnchorCount	Long	Read-only. The number of anchors in the tracing result.
Application	Application	Read-only. The Illustrator <code>Application</code> object.
AreaCount	Long	Read-only. The number of areas in the tracing result.
ImageResolution	Real	Read-only. The resolution of the source image in pixels per inch.
Parent	Object	Read-only. The object's container.
PathCount	Long	Read-only. The number of paths in the tracing result.
SourceArt	<code>PlacedItem</code> or <code>RasterItem</code>	Read-only. The raster art used to create the associated vector-art plugin group.
TracingOptions	<code>TracingOptions</code>	The options used to convert the raster artwork to vector art.
UsedColorCount	Long	Read-only. The number of colors used in the tracing result.

TracingObject methods

Method	Returns	What it does
ExpandTracing ([viewed As Boolean])	GroupItem object	<p>Converts the vector art into a new group item. The new GroupItem object replaces the PluginItem object in the document. By default, viewed is false, and the new group contains only the tracing result (the filled or stroked paths). If viewed is true, the new group retains additional information that was specified for the viewing mode, such as outlines and overlays.</p> <p>Deletes this object and its associated PluginItem object. Any group-level attributes that were applied to the plugin item are applied to the top level of the new group item.</p>
ReleaseTracing ()	PlacedItem or RasterItem object	Reverts the artwork in the document to the original source raster art and removes the traced vector art. Returns the original object used to create the tracing, and deletes this object and its associated PluginItem object.

TracingOptions

A set of options used in converting raster art to vector art by tracing.

TracingOptions properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
CornerAngle	Double	The sharpness, in degrees of a turn in the original image that is considered a corner in the tracing result path. Range: 0 to 180
Fills	Boolean	If <code>true</code> , trace with fills. At least one of <code>Fills</code> or <code>Strokes</code> must be <code>true</code> .
IgnoreWhite	Boolean	If <code>true</code> , ignores white fill color.
LivePaintOutput	boolean	If <code>true</code> , result is LivePaint art. If <code>false</code> , it is classic art. Note: A script should set this value only in preparation for a subsequent expand operation. Leaving a tracing on the artboard when this property is <code>true</code> can lead to unexpected application behavior.
MaxColors	Long	The maximum number of colors allowed for automatic palette generation. Used only if <code>TracingMode</code> is <code>color</code> or <code>grayscale</code> . Range: 2 to 256
MaxStrokeWeight	Double	The maximum stroke weight in points, when <code>Strokes</code> is <code>true</code> . Range: 0.01 to 100.0
MinArea	Long	The smallest feature, in square pixels, that is traced. For example, if it is 4, a feature of 2 pixels wide by 2 pixels high is traced.
MinStrokeLength	Double	The minimum length in pixels of features in the original image that can be stroked, when <code>Strokes</code> is <code>true</code> . Smaller features are omitted. Range: 0.0 to 200.0 Default: 20 . 0
OutputToSwatches	Boolean	If <code>true</code> , named colors (swatches) are generated for each new color created by the tracing result. Used only if <code>TracingMode</code> is 0 (<code>aiTracingModeColor</code>) or 1 (<code>aiTracingModeGray</code>).
Palette	String	The name of a color palette to use for tracing. If the empty string, use the automatic palette. Used only if <code>TracingMode</code> is 0 (<code>aiTracingModeColor</code>) or 1 (<code>aiTracingModeGray</code>).
Parent	Object	Read-only. The object's container.

Property	Value type	What it is
PathFitting	Double	The distance between the traced shape and the original pixel shape. Lower values create a tighter path fitting. Higher values create a looser path fitting. Range: 0.0 to 10.0
PreprocessBlur	Double	The amount of blur used during preprocessing, in pixels. Blurring helps reduce small artifacts and smooth jagged edges in the tracing result. Range: 0.0 to 2.0
Preset	String	Read-only. The name of a preset file containing these options.
Resample	Boolean	If <code>true</code> , resample when tracing. (This setting is not captured in a preset file.) Always <code>true</code> when the raster source art is placed or linked.
ResampleResolution	Double	The resolution to use when resampling in pixels per inch (ppi). Lower resolution increases the speed of the tracing operation. (This setting is not captured in a preset file.)
Strokes	Boolean	If <code>true</code> , trace with strokes. At least one of <code>Fills</code> or <code>Strokes</code> must be <code>true</code> . Used only if <code>TracingMode</code> is 2 ' <code>aiTracingModeBlackAndWhite</code> .
Threshold	Long	The threshold value of black-and-white tracing. All pixels with a grayscale value greater than this are converted to black. Used only if <code>TracingMode</code> is 2 ' <code>aiTracingModeBlackAndWhite</code> . Range: 0 to 255
TracingMode	AiTracingModeType	The color mode for tracing.
ViewRaster	AiViewRasterType	The view for previews of the raster image. (This setting is not captured in a preset file.)
ViewVector	AiViewVectorType	The view for previews of the vector result. (This setting is not captured in a preset file.)

TracingOptions methods

Method	Returns	What it does
LoadFromPreset (presetName as String)	Boolean	Loads a set of options from the specified preset, as found in the <code>Application.TracingPresetList</code> array.
StoreToPreset (presetName as String)	Boolean	Saves this set of options in the specified preset. Use a name found in the <code>Application.TracingPresetList</code> array, or a new name to create a new preset. For an existing preset, overwrites an unlocked preset and returns <code>true</code> . Returns <code>false</code> if the preset is locked.

Variable

A dynamic object used to create data-driven graphics. For an example, see [DataSets](#). Variables are accessed in Illustrator through the Variables palette. A variable is document-level object that can be imported or exported.

Variable properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Kind	AiVariableKind	The variable's type.
Name	String	The name of the variable.
PageItems	PageItems	Read-only. The artwork contained in the variable.
Parent	Document	Read-only. The document that contains this object.

Variable methods

Method	Returns	What it does
Delete ()	Nothing	Deletes the object.

Variables

A collection of `Variable` objects in a document.

Variables properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Document	Read-only. The object's document container.

Variables methods

Method	Returns	What it does
Add ()	Variable object	Creates a new <code>Variable</code> object.
Index (item as <code>Variable</code>)	Long	Returns the index position of the object within the collection.
Item (itemKey)	Variable object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

View

A document view in an Illustrator document, which represents a window view onto a document. Scripts cannot create new views, but can modify some properties of existing views, including the center point, screen mode, and zoom.

View properties

Property	Value type	What it is
Application	Application object	Read-only. The Illustrator <code>Application</code> object.
Bounds	Variant Array of 4 Doubles	Read-only. The bounding rectangle of this <code>View</code> relative to the current document's bounds.
CenterPoint	Variant Array of 2 Doubles	The center point of this <code>View</code> relative to the current document's bounds.
Parent	Document object	Read-only. The document that contains this object.
ScreenMode	AiScreenMode	The mode of display for this view.
Zoom	Double	The zoom factor of this view, where 100.0 is 100%.

► Setting a view

```
'Sets the view of the current document to full screen

Set appRef = CreateObject("Illustrator.Application")

Set frontDoc = appRef.activeDocument
Set viewsRef = frontDoc.Views
Set firstView = viewsRef(1)
firstView.ScreenMode = 3 'aiFullScreen
```

Views

A collection of `View` objects in a document.

Views properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Document	Read-only. The object's document container.

Views methods

Method	Returns	What it does
Index (item as View)	Long	Returns the index position of the object within the collection.
Item (itemKey)	View object	Returns an object reference to the object identified by <code>itemKey</code> (name or index).

Words

A collection of words in a text item, where each word is a `TextRange` object.

Words properties

Property	Value type	What it is
Application	Application	Read-only. The Illustrator <code>Application</code> object.
Count	Long	Read-only. Number of elements in the collection.
Parent	Object	Read-only. The object's container.

Words methods

Method	Returns	What it does
Add (contents as String [, relativeObject as TextFrame] [, insertionLocation as AiElementPlacement])	<code>TextRange</code>	Adds a new word with the specified text contents at the specified location in the current document. If location is not specified, adds the new word to the containing text frame after the current text selection or insertion point.
AddBefore (contents as String)	<code>TextRange</code>	Inserts text at the beginning of the range.
Index (itemPtr as TextRange)	Long	Returns the index position of the object within the collection.
Item (itemKey)	<code>TextRange</code>	Returns an object reference to the object identified by <code>itemKey</code> (name or index).
RemoveAll ()	Nothing	Deletes all objects in this collection.

► Counting words in a document

```
'Counts the number of words in the document and
'stores in numWords

Set appRef = CreateObject("Illustrator.Application")

If appRef.Documents.Count > 0 Then
    numWords = 0
    For Each textArt In appRef.ActiveDocument.TextFrames
        numWords = numWords + textArt.TextRange.Words.Count
    Next
End If
```

Enumerations reference

These enumeration values are defined for object properties in the VBScript dictionary.

When defining an enumerated property's value, use the numeric, rather than the text value of the enumeration. For example, the following script statement, which uses the first enumeration in the following table, `AiAlternateGlyphsForm`, specifies `aiExpert` as the constant value:

```
myCharacterAttributes.AlternateGlyphs = 2
```

Enumeration type	Values	What it means
AiAlternateGlyphsForm	aiDefaultForm = 0 aiTraditional = 1 aiExpert = 2 aiJIS78Form = 3 aiJIS83Form = 4 aiHalfWidth = 5 aiThirdWidth = 6 aiQuarterWidth = 7 aiFullWidth = 8 aiProportionalWidth = 9	
AiArtClippingOption	aiOutputArtboardBounds = 2 aiOutputArtBounds = 1 aiOutputCropRectBounds = 3	How the art are clipped during output. aiOutputArtBounds = Output size is the size of the artwork. aiOutputArtboardBounds = Output size is the size of the artboard. aiOutputCropRectBounds = Output size is the size of the drop area.
AiAutoCADColors	aiMax8Colors = 0 aiMax16Colors = 1 aiMax256Colors = 2 aiTrueColors = 3	
AiAutoCADCompatibility	aiAutoCADRelease13 = 0 aiAutoCADRelease14 = 1 aiAutoCADRelease15 = 2 aiAutoCADRelease18 = 3	
AiAutoCADExportFileFormat	aiDXF = 0 aiDWG = 1	
AiAutoCADExportOption	aiPreserveAppearance = 0 aiMaximumEditability = 1	
AiAutoCADGlobalScaleOption	aiOriginalSize = 0 aiFitArtboard = 1 aiScaleByValue = 2	
AiAutoCADRasterFormat	aiPNG = 0 aiJPEG = 1	

Enumeration type	Values	What it means
AiAutoCADUnit	aiPoints = 0 aiPicas = 1 aiInches = 2 aiMillimeters = 3 aiCentimeters = 4 aiPixels = 5	
AiAutoKernType	aiNoAutoKern = 0 aiAuto = 1 aiOptical = 2	
AiAutoLeadingType	aiBottomToBottom = 0 aiTopToTop = 1	
AiBaselineDirectionType	aiStandardBaseline = 1 aiTateChuYokoBaseline = 3 aiVerticalRotatedBaseline = 2	
AiBlendAnimationType	aiInBuild = 2 aiInSequence = 1 aiNoBlendAnimation = 0	
AiBlendModes	aiColorBlend = 14 aiColorBurn = 7 aiColorDodge = 6 aiDarken = 8 aiDifference = 10 aiExclusion = 11 aiHardLight = 5 aiHue = 12 aiLighten = 9 aiLuminosity = 15 aiMultiply = 1 aiNormalBlend = 0 aiOverlay = 3 aiSaturation = 13 aiScreen = 2 aiSoftLight = 4	The blend mode used when compositing an object.
AiBurasagariTypeEnum	aiBurasagariNone = 0 aiBurasagariStandard = 1 aiBurasagariForced = 2	
AiCaseChangeType	aiUpperCase = 0 aiLowerCase = 1 aiTitleCase = 2 aiSentenceCase = 3	
AiColor	aiColorCMYK = 1 aiColorGradient = 6 aiColorGray = 2 aiColorNone = 0 aiColorPattern = 5 aiColorRGB = 3 aiColorSpot = 4	A color space.
AiColorConversion	aiColorConversionRepurpose = 2 aiColorConversionToDest = 1 aiColorConversionNone = 0	

Enumeration type	Values	What it means
AiColorDestination	aiColorDestinationDocCMYK = 1 aiColorDestinationDocRGB = 3 aiColorDestinationProfile = 5 aiColorDestinationWorkingCMYK = 2 aiColorDestinationWorkingRGB = 4 aiColorDestinationNone = 0	
AiColorDitherMethod	aiDiffusion = &H4466736E aiNoise = &H424E6F69 aiNoReduction = &H4E6F6E65 aiPatternDither = &H5074726E	The method used to dither colors in exported GIF and PNG8 images.
AiColorModel	aiRegistration = 0 aiProcess = 1 aiSpot = 2	
AiColorProfile	aiIncludeAllProfile = 1 aiIncludeDestProfile = 4 aiIncludeRGBProfile = 3 aiLeaveProfileUnchanged = 2 aiColorProfileNone = 0	
AiColorReductionMethod	aiAdaptive = &H41647074 aiPerceptual = &H50726370 aiSelective = &H536C7476 aiWeb = &H57656220	The method used to reduce the number of colors in exported GIF and PNG8 images.
AiCompatibility	aiJapaneseVersion3 = 3 aiIllustrator8 = 8 aiIllustrator9 = 9 aiIllustrator10 = 10 aiIllustrator11 = 11 aiIllustrator12 = 12 aiIllustrator13 = 13	The version of the Illustrator file format to create when saving an EPS or Illustrator file.
AiCompressionQuality	aiJPEGHigh = 6 aiJPEGLow = 4 aiJPEGMaximum = 7 aiJPEGMedium = 5 aiJPEGMinimum = 3 aiNoCompression = 1 aiZIP4Bit = 8 aiZIP8Bit = 9 aiAutomaticJPEGMinimum = 10 aiAutomaticJPEGLow = 11 aiAutomaticJPEGMedium = 12 aiAutomaticJPEGHigh = 13 aiAutomaticJPEGMaximum = 14 aiAutomaticJPEG2000Minimum = 15 aiAutomaticJPEG2000Low = 16 aiAutomaticJPEG2000Medium = 17 aiAutomaticJPEG2000High = 18 aiAutomaticJPEG2000Maximum = 19 aiAutomaticJPEG2000Lossless = 20 aiJPEG2000Minimum = 21 aiJPEG2000Low = 22 aiJPEG2000Medium = 23 aiJPEG2000High = 24 aiJPEG2000Maximum = 25 aiJPEG2000Lossless = 26	The quality of bitmap compression used when saving a PDF file.

Enumeration type	Values	What it means
AiCropOptions	aiCropJapanese = 2 aiCropStandard = 1	The style of a document's cropping box.
AiDocumentColorSpace	aiDocumentCMYKColor = 2 aiDocumentRGBColor = 1	The color space of a document.
AiDocumentPresetType	aiBasicCMYKPreset = 5 aiBasicRGBPreset = 6 aiPrintPreset = 1 aiMobilePreset = 3 aiVideoPreset = aiWebPreset = 2	The preset types available for new documents.
AiDocumentPreviewMode	aiDefaultPreview = 1 aiPixelPreview = 2 aiOverprintPreview = 3	The document preview mode.
AiDocumentRasterResolution	aiScreenResolution = 1 aiMediumResolution = 2 aiHighResolution = 3	The preset document raster resolution.
AiDocumentTransparencyGrid	aiTransparencyGridNone = 0 aiTransparencyGridLight = 1 aiTransparencyGridMedium = 2 aiTransparencyGridDark = 3 aiTransparencyGridRed = 4 aiTransparencyGridOrange = 5 aiTransparencyGridGreen = 6 aiTransparencyGridBlue = 7 aiTransparencyGridPurple = 8	Document transparency grid colors.
AiDocumentType	aiEPS = 2 aiIllustrator = 1 aiPDF = 3	The file format used to save a file.
AiDownsampleMethod	aiNoDownsample = 0 aiAverageDownsample = 1 aiSubsample = 2 aiBicubicDownsample = 3	
AiElementPlacement	aiPlaceInside = 0 aiPlaceAtBeginning = 1 aiPlaceAtEnd = 2 aiPlaceBefore = 3 aiPlaceAfter = 4	
AiEPSPostScriptLevelEnum	aiLevel2 = 2 aiLevel3 = 3	Specifies the PostScript level to use when saving an EPS file.
AiEPSPreview	aiBWMacintosh = 2 aiBWTIFF = 4 aiColorMacintosh = 3 aiColorTIFF = 6 aiNoPreview = 1 aiTransparentColorTIFF = 5	The preview image format used when saving an EPS file.

Enumeration type	Values	What it means
AiExportType	aiGIF = 6 aiJPEG = 1 aiPhotoshop = 2 aiPNG24 = 5 aiPNG8 = 4 aiSVG = 3 aiFlash = 7 aiAutoCAD = 8	The file format used to export a file.
AiFigureStyleType	aiDefaultFigureStyle = 0 aiTabular = 1 aiTabularOldStyle = 4 aiProportional = 3 aiProportionalOldStyle = 2	
AiFlashExportStyle	aiAsFlashFile = 1 aiLayersAsFrames = 2 aiLayersAsFiles = 3 aiLayersAsSymbols = 4	
AiFlashExportVersion	aiFlashVersion1 = 1 aiFlashVersion2 = 2 aiFlashVersion3 = 3 aiFlashVersion4 = 4 aiFlashVersion5 = 5 aiFlashVersion6 = 6 aiFlashVersion7 = 7 aiFlashVersion8 = 8 aiFlashVersion9 = 9	
AiFlashImageFormat	aiLossless = 1 aiLossy = 2	
AiFlashJPEGMethod	aiJPEGStandard = 1 aiJPEGOptimized = 2	
AiFlashPlaybackSecurity	aiPlaybackLocal = 1 aiPlaybackNetwork = 2	
AiFontBaselineOption	aiNormalBaseline = 0 aiSuperScript = 1 aiSubScript = 2	
AiFontCapsOption	aiNormalCaps = 0 aiSmallCaps = 1 aiAllCaps = 2 aiAllSmallCaps = 3	
AiFontOpenTypePositionOption	aiOpenTypeDefault = 0 aiOpenTypeSuperScript = 1 aiOpenTypeSubScript = 2 aiNumerator = 3 aiDenominator = 4	
AiFontSubstitutionPolicy	aiSubstituteOblique = 0 aiSubstituteTint = 1 aiSubstituteDevice = 2	
AiGradientType	aiLinearGradient = 1 aiRadialGradient = 2	The type of the gradient, radial or linear.

Enumeration type	Values	What it means
AiImageColorSpace	aiImageCMYK = 3 aiImageGrayScale = 1 aiImageRGB = 2 aiImageLAB = 4 aiImageSeparation = 5 aiImageDeviceN = 6 aiImageIndexed = 7	The color space of a raster item or an exported Photoshop file.
AiInkPrintStatus	aiDisableInk = 0 aiEnableInk = 1 aiConvertInk = 2	
AiInkType	aiCyanInk = 0 aiMagentaInk = 1 aiYellowInk = 2 aiBlackInk = 3 aiCustomInk = 4	
AiJavaScriptExecutionMode	aiNeverShowDebugger = 1 aiDebuggerOnError = 2 aiBeforeRunning = 3	
AiJustification	aiCenter = 2 aiLeft = 0 aiRight = 1 aiFullJustifyLastLineLeft = 3 aiFullJustifyLastLineRight = 4 aiFullJustifyLastLineCenter = 5 aiFullJustify = 6	The alignment or justification for a paragraph of text.
AiKinsokuOrderEnum	aiPushIn = 0 aiPushOutFirst = 1 aiPushOutOnly = 2	
AiKnockoutState	aiDisabled = 0 aiEnabled = 1 aiInherited = 2 aiKnockoutUnknown = -1	The type of knockout to use on a PageItem.

Enumeration type	Values	What it means
AiLanguageType	aiBokmalNorwegian = 8 aiBrazillianPortuguese = 11 aiBulgarian = 20 aiCanadianFrench = 3 aiCatalan = 17 aiChinese = 29 aiCzech = 22 aiDanish = 16 aiDutch = 15 aiDutch2005Reform = 43 aiEnglish = 0 aiFinnish = 1 aiGerman2006Reform = 42 aiGreek = 25 aiHungarian = 28 aiIcelandic = 27 aiItalian = 7 aiJapanese = 30 aiNynorskNorwegian = aiOldGerman = 5 aiPolish = 23 aiRomanian = 24 aiRussian = 18 aiSpanish = 12 aiSerbian = 21 aiStandardFrench = 2 aiStandardGerman = 4 aiStandardPortuguese = aiSwedish = 13 aiSwissGerman = 6 aiSwissGerman2006Reform = 44 aiTurkish = 26 aiUKEnglish = 14 aiUkranian = 19	
AiLayerOrderType	aiTopDown = 2 aiBottomUp = 1	
AiLibraryType	aiIllustratorArtwork = 1 aiSwatches = 2 aiBrushes = 3 aiGraphicStyles = 4 aiSymbols = 5	
AiMonochromeCompression	aiCCIT3 = 1 aiCCIT4 = 2 aiMonoZIP = 3 aiNoMonoCompression = 0 aiRunLength = 4	The type of monochrome bitmap compression to use when saving a PDF.
AiOutputFlattening	aiPreserveAppearance = 1 aiPreservePaths = 0	Specifies how transparency are flattened when saving EPS and Illustrator file formats with compatibility set to versions of Illustrator less than 9.

Enumeration type	Values	What it means
AiPageItemType	aiCompoundPathItem = 1 aiGraphItem = 2 aiGroupItem = 3 aiLegacyTextItem = 11 aiMeshItem = 4 aiPathItem = 5 aiPlacedItem = 6 aiPluginItem = 7 aiRasterItem = 8 aiSymbolItem = 9 aiTextFrame = 10	The type (class) of art item that is represented by a particular PageItem.
AiPageMarksStyle	aiPageMarksRoman = 0 aiPageMarksJapanese = 1	
AiPathPointSelection	aiAnchorPoint = 2 aiLeftDirection = 3 aiLeftRightPoint = 5 aiNoSelection = 1 aiRightDirection = 4	Specifies which points, if any, of a path point are selected.
AiPDFBoxType	aiPDFArtBox = 0 aiPDFCropBox = 1 aiPDFTrimBox = 2 aiPDFBleedBox = 3 aiPDFMediaBox = 4 aiPDFBoundingBox = 5	
AiPDFChangesAllowedEnum	aiChange128None = 1 aiChange128EditPage = 2 aiChange128FillForm = 3 aiChange128Commenting = 4 aiChange128AnyChanges = 5 aiChange40None = 6 aiChange40PageLayout = 8 aiChange40Commenting = 7 aiChange40AnyChanges = 9	
AiPDFCompatibility	aiAcrobat4 = 4 aiAcrobat5 = 5 aiAcrobat6 = 6 aiAcrobat7 = 7 aiAcrobat8 = 8	The version of the Acrobat file format to create when saving a PDF file.
AiPDFOverprint	aiPreservePDFOverprint = 1 aiDiscardPDFOverprint = 2	
AiPDFPrintAllowedEnum	aiPrint128None = 1 aiPrint128LowResolution = 2 aiPrint128HighResolution = 3 aiPrint40None = 4 aiPrint40HighResolution = 5	
AiPDFTrimMarkWeight	aiTrimMarkWeight0125 = 1 aiTrimMarkWeight025 = 2 aiTrimMarkWeight05 = 3	
AiPDFXStandard	aiPDFX1A2001 = 2 aiPDFX1A2003 = 3 aiPDFX32001 = 4 aiPDFX32003 = 5 aiPDFXNone = 1	

Enumeration type	Values	What it means
AiPhotoshopCompatibility	aiPhotoshop6 = 2 aiPhotoshop8 = 1	
AiPointType	aiCorner = 2 aiSmooth = 1	The type of path point, either a curve or a corner.
AiPolarityValues	aiPositive = 1 aiNegative = -1	
AiPostScriptImageCompressionType	aiImageCompressionNone = 0 aiImageCompressionRLE = 1 aiImageCompressionJPEG = 2	
AiPrintArtworkDesignation	aiVisiblePrintableLayers = 0 aiVisibleLayers = 1 aiAllLayers = 2	
AiPrintColorIntent	aiPerceptualIntent = 0 aiSaturationIntent = 1 aiRelativeColorimetric = 2 aiAbsoluteColorimetric = 3	
AiPrintColorProfile	aiOldstyleProfile = 0 aiSourceProfile = 1 aiPrinterProfile = 2 aiCustomProfile = 3	
AiPrintColorSeparationMode	aiComposite = 0 aiOnHostSeparation = 1 aiInRIPSeparation = 2	
AiPrinterColorMode	aiColorPrinter = 0 aiGrayscalePrinter = 1 aiBlackAndWhitePrinter = 2	
AiPrinterPostScriptLevelEnum	aiPSLevel1 = 1 aiPSLevel2 = 2 aiPSLevel3 = 3	
AiPrinterTypeEnum	aiUnknownPrinterType = 0 aiPostScriptPrinter = 1 aiNonPostScriptPrinter = 2	
AiPrintFontDownloadMode	aiDownloadNone = 0 aiDownloadSubset = 1 aiDownloadComplete = 2	
AiPrintingBounds	aiArtboardBounds = 0 aiArtworkBounds = 1 aiCropBounds = 2	
AiPrintOrientation	aiPortrait = 0 aiLandscape = 1 aiReversePortrait = 2 aiReverseLandscape = 3	

Enumeration type	Values	What it means
AiPrintPosition	aiTranslateTopLeft = 1 aiTranslateTop = 2 aiTranslateTopRight = 3 aiTranslateLeft = 4 aiTranslateCenter = 5 aiTranslateRight = 6 aiTranslateBottomLeft = 7 aiTranslateBottom = 8 aiTranslateBottomRight = 9	
AiPrintTiling	aiTileSingleFullPage = 0 aiTileFullPages = 1 aiTileImageableAreas = 2	
AiRasterLinkState	aiDataFromFile = 2 aiDataModified = 3 aiNoData = 1	The status of a raster item's linked image, if the image is stored externally.
AiRulerUnits	aiUnitsCM = 3 aiUnitsInches = 2 aiUnitsMM = 6 aiUnitsPicas = 5 aiUnitsPoints = 4 aiUnitsQ = 7 aiUnitsPixels = 8 aiUnitsUnknown = 1	The default measurement units for the rulers in a document.
AiSaveOptions	aiDoNotSaveChanges = 2 aiPromptToSaveChanges = 3 aiSaveChanges = 1	Save options provided when closing a document.
AiScreenMode	aiDesktop = 2 aiFullScreen = 3 aiMultiWindow = 1	The mode of display for a view.
AiStrokeCap	aiButtEndCap = 1 aiProjectingEndCap = 3 aiRoundEndCap = 2	The type of line capping for a path stroke.
AiStrokeJoin	aiBevelEndJoin = 3 aiMiterEndJoin = 1 aiRoundEndJoin = 2	The type of joints for a path stroke.
AiStyleRunAlignmentType	aiAlignBottom = 0 aiICFBottom = 1 aiRomanBaseline = 2 aiAlignCenter = 3 aiICFTop = 4 aiAlignTop = 5	
AiSVGCSSPropertyLocation	aiEntities = 2 aiStyleAttributes = 1 aiStyleElements = 3 aiPresentationAttributes = 4	How should the CSS properties of the document be included in an exported SVG file?
AiSVGDocumentEncoding	aiASCII = 1 aiUTF16 = 3 aiUTF8 = 2	How text is encoded when exporting an SVG file.

Enumeration type	Values	What it means
AiSVGDTDVersion	aiSVG1_0 = 1 aiSVG1_1 = 2 aiSVGBasic1_1 = 5 aiSVGTiny1_1 = 3 aiSVGTiny1_1Plus = 4 aiSVGTiny1_2 = 6	SVG version compatibility for exported file.
AiSVGFontSubsetting	aiAllGlyphs = 7 aiCommonEnglish = 3 aiCommonRoman = 5 aiGlyphsUsed = 2 aiGlyphsUsedPlusEnglish = 4 aiGlyphsUsedPlusRoman = 6 aiNoFonts = 1	What font glyphs are included in an exported SVG file?
AiSVGFontType	aiCEFFont = 1 aiOutlineFont = 3 aiSVGFont = 2	Types for fonts included in exported SVG files.
AiTabStopAlignment	aiCenterTab = 1 aiDecimalTab = 3 aiLeftTab = 0 aiRightTab = 2	The alignment of a tab stop.
AiTextOrientation	aiHorizontal = 0 aiVertical = 1	The orientation of text in a text frame.
AiTextType	aiAreaText = 1 aiPathText = 2 aiPointText = 0	The type of text art displayed by this object.
AiTracingModeType	aiTracingModeBlackAndWhite = 2 aiTracingModeColor = 0 aiTracingModeGray = 1	
AiTransformation	aiTransformBottom = 7 aiTransformBottomLeft = 4 aiTransformBottomRight = 10 aiTransformCenter = 6 aiTransformDocumentOrigin = 1 aiTransformLeft = 3 aiTransformRight = 9 aiTransformTop = 5 aiTransformTopLeft = 2 aiTransformTopRight = 8	The point to use as the anchor point about which an object is rotated, resized or transformed.
AiTrappingType	aiNormalTrapping = 0 aiTransparent = 1 aiOpaque = 2 aiIgnoreOpaque = 3	
AiUserInteractionLevel	aiDontDisplayAlerts = -1 aiDisplayAlerts = 2	
AiVariableKind	aiGraph = 5 aiImage = 4 aiTextual = 3 aiUnknownKind = 1 aiVisibility = 2	The enumerated type of the kind of variable.

Enumeration type	Values	What it means
AiViewRasterType	aiTracingViewRasterAdjustedImage = 2 aiTracingViewRasterNoImage = 0 aiTracingViewRasterOriginalImage = 1 aiTracingViewRasterTransparentImage = 3	The raster viewing mode for tracing.
AiViewVectorType	aiTracingViewVectorNoTracingResult = 0 aiTracingViewVectorOutlines = 2 aiTracingViewVectorOutlinesWithTracing = 3 aiTracingViewVectorTracingResult = 1	The vector viewing mode for tracing.
AiWariChuJustificationType	aiWariChuLeft = 0 aiWariChuRight = 1 aiWariChuCenter = 2 aiWariChuFullJustifyLastLineLeft = 3 aiWariChuFullJustifyLastLineRight = 4 aiWariChuFullJustifyLastLineCenter = 5 aiWariChuAutoJustify = 6	
AiZOrderMethod	aiBringForward = 2 aiBringToFront = 1 aiSendBackward = 3 aiSendToBack = 4	The method used to arrange an art item's position in the stacking order of its parent group or layer, as specified with the ZOrder method.