

ADOBE®

ILLUSTRATOR® CC

2017

ADOBE ILLUSTRATOR CC 2017

**脚本参考：
JAVASCRIPT**



© 2016 Adobe系统公司。版权所有。

Adobe Illustrator CC 2017脚本参考: JavaScript

如果本指南与包含最终用户协议的软件一起分发，则本指南以及其中描述的软件均已获得许可，并且只能根据该许可的条款使用或复制。除非获得任何此类许可的许可，否则未经Adobe事先书面许可，不得以任何形式或手段（电子、机械、记录或以其他方式）复制、存储或在本检索系统中传播本指南的任何部分。系统公司。请注意，即使未随包含最终用户许可协议的软件一起分发，本指南中的内容也受版权法保护。

本指南的内容仅供参考，如有更改，恕不另行通知，并且不应解释为Adobe Systems Incorporated的承诺。Adobe Systems Incorporated对本指南中信息内容中可能出现的任何错误或不准确性不承担任何责任。

请记住，您可能要包含在项目中的现有图稿或图像可能受到版权法的保护。未经授权将此类材料合并到您的新作品中可能会侵犯版权所有者的权利。请确保获得版权所有者的任何许可。

示例模板中对公司名称的任何引用仅出于演示目的，并不旨在引用任何实际组织。

Adobe, Adobe徽标, Acrobat, Flash, Illustrator, Macromedia和Photoshop是Adobe Systems Incorporated在美国和/或其他国家的注册商标或商标。

JavaScript和所有与Java相关的标记是Sun Microsystems, Incorporated在美国和其他国家/地区的商标或注册商标。

所有其他商标均为其各自所有者的财产。

Adobe系统公司，美国加利福尼亚州95110，圣何塞，公园大道345号。致美国政府最终用户的通知。软件和文档是“商业项目”，该术语在48 C.F.R.§2.101，由“商用计算机软件”和“商用计算机软件文档”组成，此类术语在48 C.F.R.§12.212或48 C.F.R.第227.7202条，如适用。符合48 C.F.R.§12.212或48 C.F.R.§§227.7202-1至227.7202-4，如适用，商用计算机软件和商用计算机软件文档已被许可给美国政府最终用户（a）仅作为商业项目，并且（b）仅授予所有人其他最终用户（根据此处的条款和条件）。根据美国版权法保留未发布的权利。Adobe系统公司，美国加利福尼亚州圣何塞，公园大道345号，邮政编码：95110-2704。对于美国政府最终用户，Adobe同意遵守所有适用的机会均等法律，包括（如果适用）经修订的1974年越南越战退伍军人调整协助法（38 USC 4212）第402条行政命令的规定，以及经修正的1973年《康复法案》第503节，以及41 CFR第60-1至60-60、60-250和60-741部分的规定。前一句中包含的平权行动条款和规定应通过引用并入。

Contents

1	JavaScript Object Reference	7
	Application	8
	Artboard	13
	Artboards	14
	Brush	15
	Brushes	16
	CharacterAttributes	17
	Characters	21
	CharacterStyle	22
	CharacterStyles	23
	CMYKColor	25
	Color	26
	CompoundPathItem	27
	CompoundPathItems	31
	Dataset	32
	Datasets	34
	Document	35
	DocumentPreset	44
	Documents	45
	EPSSaveOptions	46
	ExportOptionsAutoCAD	48
	ExportOptionsFlash	50
	ExportOptionsGIF	52
	ExportOptionsJPEG	54
	ExportOptionsPhotoshop	56
	ExportOptionsPNG8	58
	ExportOptionsPNG24	60
	ExportOptionsSVG	61
	ExportOptionsTIFF	63
	FXGSaveOptions	64
	Gradient	65
	GradientColor	67
	Gradients	68
	GradientStop	69

GradientStops 70

GraphicStyle 72

GraphicStyles 73

GraphItem 74

GraphItems 77

GrayColor 78

GroupItem 79

GroupItems 83

IllustratorSaveOptions 84

ImageCaptureOptions 86

Ink 87

InkInfo 88

InsertionPoint 89

InsertionPoints 90

LabColor 91

Layer 92

Layers 95

LegacyTextItem 96

LegacyTextItems 99

Lines 100

Matrix 101

MeshItem 102

MeshItems 105

NoColor 106

NonNativeItem 107

NonNativeItems 110

OpenOptions 111

OpenOptionsAutoCAD 112

OpenOptionsFreeHand 113

OpenOptionsPhotoshop 114

PageItem 115

PageItems 118

Paper 120

PaperInfo 121

ParagraphAttributes 122

Paragraphs 126

ParagraphStyle 127

ParagraphStyles 128

PathItem 130

PathItems 135

PathPoint 137

PathPoints 138

Pattern 139

PatternColor 140

Patterns 142

PDFFileOptions 143

PDFSaveOptions 144

PhotoshopFileOptions 150

PlacedItem 151

PlacedItems 155

PluginItem 156

PluginItems 159

PPDFile 160

PPDFileInfo 161

Preferences 163

PrintColorManagementOptions 165

PrintColorSeparationOptions 166

PrintCoordinateOptions 167

Printer 169

PrinterInfo 170

PrintFlattenerOptions 172

PrintFontOptions 174

PrintJobOptions 175

PrintOptions 177

PrintPageMarksOptions 179

PrintPaperOptions 180

PrintPostScriptOptions 181

RasterEffectOptions 182

RasterItem 183

RasterItems 187

RasterizeOptions 189

RGBColor 190

Screen 191

ScreenInfo 192

ScreenSpotFunction 193

Spot 194

SpotColor 196

Spots 197

Story 199

Stories 201

Swatch 202

Swatches 203

SwatchGroup 204

SwatchGroups 205

Symbol 206

SymbolItem 207

SymbolItems 210

Symbols 211

TabStopInfo 213

Tag 214

Tags 216

TextFont 217

TextFonts 218

TextFrameItem 220

TextFrameItems 224

TextPath 226

TextRange 228

TextRanges 230

TracingObject 231

TracingOptions 233

Variable 235

Variables 236

View 237

Views 238

Words 239

2 Scripting Constants 241

1 JavaScript对象参考

本节介绍类型库中的所有对象类。每个类列表包括以下内容：

- ▶ **class**类的属性，包括值类型，只读状态和说明。
- ▶ 该类的方法。该方法所需的常量和值类型以粗体显示。必填项以朴素的形式显示。方括号[]包围的所有项目都是可选的。
- ▶ 说明特殊问题的注释。
- ▶ 1示例代码可帮助说明对象类的语法和典型的工作流用法。
这些示例旨在清晰地演示语法，而不显示构造JavaScript脚本的最佳或最有效方法。例如，错误检查通常是简短的或丢失的。但是，可以将示例进行组合和扩展，以使脚本具有更大的功能。

Application

使用预定义的全局应用程序对象引用的Adobe®Illustrator®应用程序对象，其中包含所有其他Illustrator对象。

Application properties

属性	值类型	注释
<code>activeDocument</code>	Document	Illustrator中的活动（最前面）文档。
<code>browserAvailable</code>	boolean	只读。如果为true，则可以使用Web浏览器。
<code>buildNumber</code>	string	只读。该应用程序的内部版本号。
<code>colorSettingsList</code>	object	只读。当前可供使用的颜色设置文件列表。
<code>coordinateSystem</code>	CoordinateSystem	当前正在使用的坐标系，文档或画板。
<code>defaultColorSettings</code>	File	只读。当前应用程序语言环境的默认颜色设置文件。
<code>documents</code>	Documents	只读。应用程序中的文档。
<code>flattenerPresetList</code>	object	只读。当前可以使用的拼合器样式名称列表。
<code>freeMemory</code>	number (long)	只读。Illustrator分区内未使用的内存量（以字节为单位）。
<code>locale</code>	string	只读。应用程序的语言环境。
<code>name</code>	string	只读。应用程序的名称（与应用程序文件的文件名无关）。
<code>pasteRememberLayers</code>	boolean	只读。如果为true，则粘贴操作将维持图层结构。
<code>path</code>	File	只读。应用程序的文件路径。
<code>PDFPresetsList</code>	object	只读。可供使用的预设PDF选项名称列表。
<code>PPDFileList</code>	object	只读。当前可用的PPD文件列表。
<code>preferences</code>	Preferences	Illustrator的首选项设置。
<code>printerList</code>	array of Printer	只读。已安装打印机的列表。
<code>printPresetsList</code>	object	只读。可供使用的预设打印选项名称列表。
<code>scriptingVersion</code>	string	只读。脚本插件的版本。

属性	值类型	注释
<code>selection</code>	array of objects	活动（最前面）文档中所有当前选定的对象。
<code>startupPresetsList</code>	object	只读。可用于创建新文档的预设列表。
<code>textFonts</code>	TextFonts	只读。安装的字体。
<code>tracingPresetList</code>	array of string	只读。可用的预设跟踪选项名称列表。
<code>typename</code>	string	只读。被引用对象的类名。
<code>userInteractionLevel</code>	UserInteractionLevel	处理脚本命令时，应允许与用户进行何种级别的交互。
<code>version</code>	string	只读。该应用程序的版本。
<code>visible</code>	boolean	只读。如果为true，则该应用程序可见。

Application methods

方法	参数类型	返回值	作用
<code>beep</code> ()		nothing	提醒用户。
<code>concatenateMatrix</code> (matrix, secondMatrix)	Matrix Matrix	Matrix	将两个矩阵连接在一起。
<code>concatenateRotationMatrix</code> (matrix, angle)	Matrix number (double)	Matrix	将旋转平移连接到变换矩阵。
<code>concatenateScaleMatrix</code> (matrix [, scaleX] [, scaleY])	Matrix number (double) number (double)	Matrix	将比例转换连接到转换矩阵。
<code>concatenateTranslationMatrix</code> (matrix [, deltaX] [, deltaY])	Matrix number (double) number (double)	Matrix	将翻译加入转换矩阵。
<code>convertSampleColor</code> (sourceColorSpace, sourceColor, destColorSpace, colorConvertPurpose [, sourceHasAlpha] [, destHasAlpha])	ImageColorSpace ColorComponents ImageColorSpace ColorConvertPurpose boolean boolean	array of ColorComponents	将样本分量颜色从一种颜色空间转换为另一种颜色空间。
<code>copy</code> ()		nothing	将当前选择复制到剪贴板。

方法	参数类型	返回值	作用
<code>cut</code> ()		nothing	将当前选择剪切到剪贴板。
<code>deleteWorkspace()</code> (workspaceName)	string	Boolean	删除现有的工作空间。
<code>getIdentityMatrix</code> ()		Matrix	返回一个单位矩阵。
<code>getIsFileOpen</code> (filePath)	string	Boolean	返回指定的文件路径是否打开
<code>getPPDFileInfo</code> (name)	string	PPDFileInfo	获取指定PPD文件的详细文件信息。
<code>getPresetFileOfType</code> (presetType)	DocumentPresetType	File	返回指定预设类型的应用程序默认文档配置文件的完整路径。
<code>getPresetSettings</code> (preset)	string	DocumentPreset	使用给定的预设名称从模板中检索跟踪选项设置。
<code>getRotationMatrix</code> ([angle])	number (double)	Matrix	返回包含一个旋转的变换矩阵。 注意： 需要一个以度为单位的值。例如，30将对象逆时针旋转30度；-30将对象顺时针旋转30度。
<code>getScaleMatrix</code> ([scaleX] [, scaleY])	number (double) number (double)	Matrix	返回包含单个比例的转换矩阵。 注意： 需要一个百分比值。例如，60将对象缩放到其原始大小的60%；200会使对象的边界加倍。
<code>getScriptableHelpGroup</code> ()		variant	获取表示应用栏中的搜索小部件的可编写脚本的帮助组对象。
<code>getTranslationMatrix</code> ([deltaX] [, deltaY])	number (double) number (double)	Matrix	返回包含单个转换的转换矩阵。 注意： 需要一个以磅为单位的值。例如，({100,200})将对象向右移动100 pt，向上移动200 pt；在每个数字之前向左和向下移动对象的负号。

方法	参数类型	返回值	作用
invertMatrix (matrix)	Matrix	Matrix	反转矩阵。
isEqualMatrix (matrix, secondMatrix)	Matrix Matrix	boolean	检查两个矩阵是否相等。
isSingularMatrix (Matrix)	Matrix	boolean	检查矩阵是否为奇异且不能反转。
loadColorSettings (fileSpec)	File	nothing	从指定的文件加载颜色设置，或者如果文件为空，则关闭颜色管理。
open (file [, documentColorSpace] [, options])	File DocumentColorSpace anything	Document	O 写入指定的文档文件。如果您打开同时包含RGB和CMYK颜色的Illustrator 9之前的文档，并且提供了documentColorSpace，则所有颜色都将转换为指定的颜色空间。如果未提供该参数，则Illustrator将打开一个对话框，以便用户可以选择颜色空间。
paste()		nothing	将当前剪贴板内容粘贴到当前文档中。
quit ()		nothing	退出Illustrator。请注意，如果剪贴板中包含数据，则Illustrator可能会显示一个对话框，提示用户为其他应用程序保存数据。
redo()		nothing	重做最近未完成的事务。
redraw ()		nothing	强制Illustrator重新绘制其所有窗口。
resetWorkspace() ()		Boolean	重置当前工作空间。
saveWorkspace() (workspaceName)	string	Boolean	保存一个新的工作区。
sendScriptMessage (pluginName, messageSelector, inputString)	string string string	string	使用给定的输入参数将插件定义的命令消息发送到插件，并返回插件定义的结果字符串。
showPresets (fileSpec)	File	PrintPresetList	从文件获取预设。

方法	参数类型	返回值	作用
switchWorkspace() (workspaceName)	string	Boolean	切换到指定的工作空间。
translatePlaceholderText (text)	string	string	将占位符文本转换为常规文本（一种以十六进制值输入Unicode点的方法）。
undo()		nothing	撤消最近的操作。

复制活动文档

//将活动文档中的所有选定项目复制到新文档中。

```
var newItem;
var docSelected = app.activeDocument.selection;

if ( docSelected.length > 0 ) {
    //创建一个新文档，并将选定的项目移到其中。
    var newDoc = app.documents.add();
    if ( docSelected.length > 0 ) {
        for ( i = 0; i < docSelected.length; i++ ) {
            docSelected[i].selected = false;
            newItem = docSelected[i].duplicate( newDoc,
                ElementPlacement.PLACEATEND );
        }
    }
    else {
        docSelected.selected = false;
        newItem = docSelected.parent.duplicate( newDoc,
            ElementPlacement.PLACEATEND );
    }
}
else {
    alert( "请选择一个或多个图稿" );
}
```

Artboard

画板对象代表文档中的单个画板。一个文档中可以有1至100个画板。

Artboard properties

属性	值类型	注释
<code>artboardRect</code>	<code>rect</code>	画板的大小和位置。
<code>name</code>	<code>string</code>	画板的唯一标识名称。
<code>parent</code>	Document	只读。此对象的父级。
<code>rulerOrigin</code>	<code>Point</code>	画板的标尺原点，相对于画板的左上角。
<code>rulerPAR</code>	<code>number (double)</code>	像素长宽比，如果标尺是像素，则用于标尺可视化。范围：0.1至10.0
<code>showCenter</code>	<code>boolean</code>	显示中心标记。
<code>showCrossHairs</code>	<code>boolean</code>	显示十字线。
<code>showSafeAreas</code>	<code>boolean</code>	显示标题和动作安全区域（用于视频）。
<code>typename</code>	<code>string</code>	只读。该对象的类名称。

Artboards methods

方法	参数类型	返回值	作用
<code>remove</code> ()		Nothing	删除此画板对象。您无法删除文档中的最后一个画板。

Artboards

A collection of `Artboard` objects.

Artboards properties

属性	值类型	作用
<code>length</code>	number	只读。集合中的数据集合数量
<code>parent</code>	Artboard	只读。包含此数据集的对象的名称
<code>typename</code>	string	只读。被引用对象的类名。

Artboards methods

方法	参数类型	返回值	作用
<code>add</code> (artboardRect)	rect	Artboard	创建一个新的 <code>Artboard</code> 对象。
<code>getActiveArtboardIndex</code> ()		number (long)	检索活动画板在文档列表中的索引位置。返回从0开始的索引。
<code>getByName</code> (name)	string	Artboard	获取具有指定名称的集合中的第一个元素。
<code>insert</code> (artboardRect, index)	rect number (long)	Nothing	创建一个新的 Artboard 对象，并将其插入列表中的给定索引处。
<code>remove</code> (index)	number (long)	Nothing	删除给定索引的画板对象。您无法删除文档中的最后一个画板。
<code>setActiveArtboardIndex</code> (index)	number (long)	Nothing	使给定索引处的画板处于活动状态，并使其以迭代顺序为当前状态。

Brush

Illustrator文档中的画笔。画笔包含在文档中。用户可以在Illustrator中创建其他画笔。您可以在脚本中访问画笔，但不能创建它们。

Brush properties

属性	值类型	注释
name	string	画笔的名称。
parent	Document	只读。包含此笔刷的文档。
typename	string	只读。被引用对象的类名。

Brush methods

方法	参数类型	返回值	作用
applyTo (artItem)	PageItem	Nothing	将画笔应用于特定的图稿。

应用一个画笔

//复制并分组当前选择中的所有项目，然后将相同的笔刷应用于组中的每个项目

```

if ( app.documents.length > 0 ) {

    docSelection = app.activeDocument.selection;
    if ( docSelection.length > 0 ) {
        newGroup = app.activeDocument.groupItems.add();

        for ( i = 0; i < docSelection.length; i++ ) {
            newItem = docSelection[i].duplicate();
            newItem.moveToBeginning( newGroup );
        }
        brush4 = app.activeDocument.brushes[1];
        brush4.applyTo( newGroup );
    }
}

```

Brushes

文档中画笔对象的集合。

Brushes properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。包含此笔刷集合的文档。
typename	string	只读。被引用对象的类名。

Brushes methods

方法	参数类型	返回值	作用
getByName (name)	string	Brush	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	Brush	从集合中获取一个元素。

画笔计数

```
//计算活动文档中的所有笔刷  
  
if ( app.documents.length > 0 ) {  
    numberOfBrushes = app.activeDocument.brushes.length;  
}
```


CharacterAttributes

指定文本框中包含的字符的属性。`CharacterStyle`对象通过其`characterAttributes`属性将这些属性与特定的文本范围相关联。

注意：字符属性没有默认值，并且在明确设置之前是不确定的。

CharacterAttributes properties

属性	值类型	注释
<code>akiLeft</code>	number (double)	要添加到字符左侧的字符间距，以千分之一em为单位（在完全对齐时，该间距不会压缩或扩展）。
<code>akiRight</code>	number (double)	要添加到字符右侧的字符间距，以千分之一em为单位（在完全对齐时，该间距不会压缩或扩展）。
<code>alignment</code>	StyleRunAlignmentType	字符对齐类型。
<code>alternateGlyphs</code>	AlternateGlyphsForm	替代字形形式。
<code>autoLeading</code>	boolean	如果为true，则应使用自动前导。
<code>baselineDirection</code>	BaselineDirectionType	日语文本基线方向。
<code>baselinePosition</code>	FontBaselineOption	文本的基线位置。
<code>baselineShift</code>	number (double)	文本基线的点偏移量。
<code>capitalization</code>	FontCapsOption	文字大小写。
<code>connectionForms</code>	boolean	如果为true，则应使用OpenType®连字形式。
<code>contextualLigature</code>	boolean	如果为true，则应使用上下文连字。
<code>discretionaryLigature</code>	boolean	如果为true，则应使用任意连字。
<code>figureStyle</code>	FigureStyleType	OpenType字体中的数字样式。
<code>fillColor</code>	Color	文本的颜色填充。

属性	值类型	注释
fractions	boolean	如果为true, 则应使用OpenType分数。
horizontalScale	number (double)	字符水平缩放系数以百分比表示 (100 = 100%)。
italics	boolean	如果为true, 则日语OpenType字体支持斜体。
kerningMethod	AutoKernType	使用自动字距调整方法。
language	LanguageType	文字语言。
leading	number (double)	两行文本之间的间距, 以磅为单位。
ligature	boolean	如果为true, 则应使用连字。
noBreak	boolean	如果为true, 则不允许换行。
openTypePosition	FontOpenTypePositionOption	OpenType基线位置。
ordinals	boolean	如果为true, 则应使用OpenType序号。
ornaments	boolean	如果为true, 则应使用OpenType修饰。
overprintFill	boolean	如果为true, 则文本的填充为套印。
overprintStroke	boolean	如果为true, 则文本的描边为套印。
parent	object	只读。对象的容器。
proportionalMetrics	boolean	如果为true, 则日语OpenType字体支持比例字形。
rotation	number (double)	字符旋转角度 (以度为单位)。
size	number (double)	字体大小 (以磅为单位)。
strikeThrough	boolean	如果为true, 则字符使用删除线样式。
strokeColor	Color	文字笔触的颜色。
strokeWeight	number (double)	描边的线宽。

属性	值类型	注释
stylisticAlternates	boolean	如果为true, 则应使用OpenType样式替代项。
swash	boolean	如果为true, 则应使用OpenType swash。
tateChuYokoHorizontal	number (long)	Tate-Chu-Yoko水平调整的分 数。
tateChuYokoVertical	number (long)	Tate-Chu-Yoko垂直调整 点数。
textFont	TextFont	可使用的字体列表。
titling	boolean	如果为true, 则应使用 OpenType标题替代。
tracking	number (long)	跟踪或距离字距调整量, 以 em的千分之一为单位。
Tsume	number (double)	日语字符周围的空间减少百 分比。
typename	string	只读。对象的类名称。
underline	boolean	如果为true, 则在字 符下划线。
verticalScale	number (double)	字符垂直缩放比例, 以百分比 表示 (100 = 100%)。
wariChuCharactersAfterBreak	number (long)	指定Wari-Chu文本 (日文插 入插图) 中的字符如何分成 两行或更多行。
wariChuCharactersBeforeBreak	number (long)	指定Wari-Chu文本 (日文插 入插图) 中的字符如何分成 两行或更多行。
wariChuEnabled	boolean	如果为true, 则启用Wari-Chu。
wariChuJustification	WariChuJustificationType	Wari-Chu理由。
wariChuLineGap	number (long)	Wari-Chu线间隙。
wariChuLines	number (long)	Wari-Chu (多行文本适合 一个空格) 的行数。
wariChuScale	number (double)	Wari-Chu量表。

设置字符属性

//创建一个新文档，添加一个简单的文本项，然后递增地增加每个字符的水平和/或垂直比例属性

```
var docRef = documents.add();
var textRef = docRef.textFrames.add();
textRef.contents = "I Love Scripting!";
textRef.top = 400;
textRef.left = 100;

//逐步增加每个字符的比例
var charCount = textRef.textRange.characters.length;
var size = 100;
for(i=0; i<charCount; i++, size *= 1.2) {

    textRef.textRange.characters[i].characterAttributes.horizontalScale
        = size;
    textRef.textRange.characters[i].characterAttributes.verticalScale
        = size;
}
```

Characters

字符的集合（长度为1的TextRange对象）。元素未命名；您必须按索引访问它们。

Characters properties

属性	值类型	注释
length	number	只读。集合中的字符数。
parent	object	只读。包含此字符的文字图稿作品。
typename	string	只读。被引用对象的类名。

Characters methods

方法	参数类型	返回值	作用
add (contents [,relativeObject] [,insertionLocation])	string TextFrameItem ElementPlacement	TextRange	在当前文档中的指定位置添加具有指定文本内容的新字符。如果未指定位置，则将新字符添加到当前文本选择或插入点之后的包含文本框中。
addBefore (contents)	string	TextRange	在指定的文本选择之前添加一个字符。
index (itemKey)	number	TextRange	从集合中获取一个元素。
removeAll ()		Nothing	删除集合中的所有元素。

计数字符

//计算活动文档中的所有字符，包括空格，并存储在numChars中

```
if ( app.documents.length > 0 ) {
    var doc = app.activeDocument;
    var numChars = 0;
    for ( i = 0; i < doc.textFrames.length; i++ ) {
        textArtRange = doc.textFrames[i].contents;
        numChars += textArtRange.length;
    }
}
```

CharacterStyle

将字符属性与字符相关联。有关示例，请参见 [CharacterStyles](#)。

CharacterStyle properties

属性	值类型	注释
<code>characterAttributes</code>	CharacterAttributes	只读。样式的字符属性。
<code>name</code>	string	字体名称。
<code>parent</code>	object	只读。对象的容器。
<code>typename</code>	string	只读。对象的类名称。

CharacterStyle methods

方法	参数类型	返回值	作用
<code>applyTo</code> (textItem [, clearingOverrides])	object boolean	Nothing	将字符样式应用于文本对象。
<code>remove</code> ()		Nothing	删除对象。

CharacterStyles

CharacterStyle对象的集合。

CharacterStyles properties

属性	值类型	注释
length	number	只读。集合中元素的数量。
parent	object	只读。对象的容器。
typename	string	只读。对象的类名称。

CharacterStyles methods

方法	参数类型	返回值	作用
add (name)	string	CharacterStyle	创建命名的字符样式。
getByName (name)	string	CharacterStyle	使用提供的名称获取集合中的第一个元素。
index (itemKey)	string, number	CharacterStyle	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

使用字符样式

```
// 在新文档中创建3个文本框架，然后创建
// 字符样式并将其应用于每个文本框架。

var docRef = documents.add();
var textRef1 = docRef.textFrames.add();
textRef1.contents = "Scripting is fun!";
textRef1.top = 700;
textRef1.left = 50;

var textRef2 = docRef.textFrames.add();
textRef2.contents = "Scripting is easy!";
textRef2.top = 625;
textRef2.left = 100;

var textRef3 = docRef.textFrames.add();
textRef3.contents = "Everyone should script!";
textRef3.top = 550;
textRef3.left = 150;
redraw();

// 创建新的字符样式
var charStyle = docRef.characterStyles.add("BigRed");
```

```
// 设置字符属性
var charAttr = charStyle.characterAttributes;
charAttr.size = 40;
charAttr.tracking = -50;
charAttr.capitalization = FontCapsOption.ALLCAPS;
var redColor = new RGBColor();
redColor.red = 255;
redColor.green = 0;
redColor.blue = 0;
charAttr.fillColor = redColor;

// 应用于文档中的每个textFrame
charStyle.applyTo(textRef1.textRange);
charStyle.applyTo(textRef2.textRange);
charStyle.applyTo(textRef3.textRange);
```


CMYKColor

CMYK颜色规范，用于需要颜色对象的地方。

如果文档的颜色空间是RGB，并且您使用CMYK为该文档中的页面项目指定了颜色值，则Illustrator会将CMYK颜色规范转换为RGB颜色规范。如果文档的颜色空间为CMYK，而您使用RGB指定颜色，则会发生相同的情况。由于这种翻译可能会丢失信息，因此您应该使用与文档的实际颜色空间匹配的类型来指定颜色。

CMYKColor properties

属性	值类型	注释
black	number (double)	黑色值。范围0.0–100.0。默认值：0.0
cyan	number (double)	青色值。范围0.0–100.0。默认值：0.0
magenta	number (double)	洋红色值。范围0.0–100.0。默认值：0.0
typename	string	只读。被引用对象的类名。
yellow	number (double)	黄色值。范围0.0–100.0。默认值：0.0

Setting a CMYK color

```
// 设置最前面的路径项的填充颜色
// 当前文档为浅紫色CMYK颜色

if ( app.documents.length > 0 && app.activeDocument.pathItems.length > 0 ) {
    frontPath = app.activeDocument.pathItems[0];
    //设置CMYK对象的颜色值
    newCMYKColor = new CMYKColor(); newCMYKColor.black = 0;
    newCMYKColor.cyan = 30.4;
    newCMYKColor.magenta = 32;
    newCMYKColor.yellow = 0;
    //在路径项中使用颜色对象
    frontPath.filled = true;
    frontPath.fillColor = newCMYKColor;
}
}
```

Color

Illustrator中使用的所有颜色类的抽象父类。子类为：

[CMYKColor](#)
[GradientColor](#)
[GrayColor](#)
[LabColor](#)
[NoColor](#)
[PatternColor](#)
[RGBColor](#)
[SpotColor](#)

CompoundPathItem

复合路径。这些对象由多个相交的路径组成，从而在透明的内部空间中各组成部分路径重叠。`pathItems`属性提供对组成复合路径的路径的访问。

当脚本询问文档中包含的路径时，包含在文档中复合路径或组中的路径将作为单独路径返回。但是，当脚本要求包含复合路径或组的层中的路径时，不会返回复合路径或组中包含的路径。

复合路径中的所有路径共享属性值。因此，如果您设置了复合路径中任一路径的属性值，则所有其他组件路径的属性都将使用新值进行更新。

CompoundPathItem properties

属性	值类型	注释
<code>artworkKnockout</code>	KnockoutState	此对象用于创建蒙版对象吗？如果是，则使用哪种蒙版对象。
<code>blendingMode</code>	BlendModes	合成对象时使用的模式。
<code>controlBounds</code>	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
<code>editable</code>	boolean	只读。如果为 <code>true</code> ，则此项目是可编辑的。
<code>geometricBounds</code>	array of 4 numbers	只读。除笔划宽度以外的对象边界。
<code>height</code>	number (double)	复合路径项目的高度（不包括笔划宽度）。
<code>hidden</code>	boolean	如果为 <code>true</code> ，则此复合路径项被隐藏。
<code>isIsolated</code>	boolean	如果为 <code>true</code> ，则隔离此对象。
<code>layer</code>	Layer	只读。该复合路径项所属的层。
<code>left</code>	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。
<code>locked</code>	boolean	如果为 <code>true</code> ，则此复合路径项被锁定。
<code>name</code>	string	此复合路径项的名称。
<code>note</code>	string	分配给该项目的注释。
<code>opacity</code>	number (double)	对象的不透明度。范围：0.0至100.0
<code>parent</code>	Layer or GroupItem	只读。此对象的父级。
<code>pathItems</code>	PathItems	只读。此复合路径中的路径集合。

属性	值类型	注释
position	array of 2 numbers	compoundPathItem 对象的左上角的位置（以磅为单位），格式为[x, y]。不包括行程重量。
selected	boolean	如果为true，则选择此复合路径项。
sliced	boolean	如果为true，则将切片。默认值: false
tags	Tags	只读。此对象中包含的标签。
top	number (double)	项目顶部的位置（以磅为单位，从页面底部开始测量）。
typename	string	只读。被引用对象的类名。
uRL	string	分配给此复合路径项的Adobe URL标记的值。
visibilityVariable	Variant	绑定到项目的可见性变量。
visibleBounds	array of 4 numbers	只读。复合路径项目的可见边界，包括笔触宽度。
width	number (double)	复合路径项目的宽度，不包括笔划宽度。
wrapInside	boolean	如果为true，则应将文本框架对象包装在该对象内。
wrapOffset	number (double)	在此对象上环绕文字时要使用的偏移量。
wrapped	boolean	如果为true，则将文本框架对象包装在该对象周围（文本框架必须在该对象上方）。
zOrderPosition	number (long)	只读。该图稿的位置在包含该图稿的组或图层（父对象）的堆叠顺序中。

CompoundPathItem methods

方法	参数类型	返回值	作用
duplicate ([relativeObject] [,insertionLocation])	object ElementPlacement	CompoundPathItem	创建所选对象的副本。
move (relativeObject, insertionLocation)	object ElementPlacement	Nothing	移动对象。
remove ()		Nothing	删除该对象。
resize (scaleX, scaleY [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,scaleAbout])	number (double) number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	缩放图稿，其中 scaleX 是水平缩放因子，而 scaleY 是垂直缩放因子。100.0 = 100%。
rotate (angle [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,rotateAbout])	number (double) boolean boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转图稿。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象。
transform (transformationMatrix [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,transformAbout])	Matrix boolean boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换图稿。
translate ([deltaX] [,deltaY] [,transformObjects] [,transformFillPatterns] [,transformFillGradients] [,transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新放置图稿，其中 delX 是水平偏移，而 delY 是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列图稿的位置。

在文档中选择路径

```
//选择不属于复合路径的所有路径if
( app.documents.length > 0 ) {

    doc = app.activeDocument;
    count = 0;
    if ( doc.pathItems.length > 0 ) {
        thePaths = doc.pathItems;
        numPaths = thePaths.length;
        for ( i = 0; i < doc.pathItems.length; i++ ) {
            pathArt = doc.pathItems[i];
            if ( pathArt.parent.typename != "CompoundPathItem" ) {
                pathArt.selected = true;
                count++;
            }
        }
    }
}
```

创建和修改复合路径项

//创建一个包含3个路径的新复合路径项目，然后为复合路径中的所有项目设置笔触的宽度和/或颜色

```
if ( app.documents.length > 0 ) {

    doc = app.activeDocument;
    newCompoundPath = doc.activeLayer.compoundPathItems.add();

    // Create the path items
    newPath = newCompoundPath.pathItems.add();
    newPath.setEntirePath( Array( Array(30, 50), Array(30, 100) ) );

    newPath = newCompoundPath.pathItems.add();
    newPath.setEntirePath( Array( Array(40, 100), Array(100, 100) ) );

    newPath = newCompoundPath.pathItems.add();
    newPath.setEntirePath( Array( Array(100, 110), Array(100, 300) ) );

    // Set stroke and width properties of the compound path
    newPath.stroked = true;
    newPath.strokeWidth = 3.5;
    newPath.strokeColor = app.activeDocument.swatches[3].color;
}
```

CompoundPathItems

CompoundPathItem对象的集合。

CompoundPathItem methods

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此集合的父级（Layer或GroupItem）。
typename	string	只读。被引用对象的类名。

CompoundPathItem methods

方法	参数类型	返回值	作用
add ()		CompoundPathItem	创建一个新的CompoundPathItem。
getByName (name)	string	CompoundPathItem	获取具有指定名称的集合中的一个元素。
index (itemKey)	string, number	CompoundPathItem	获取具有指定名称的集合中的一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

统计复合路径

```
//计算当前文档第1层中的所有复合路径项
if ( app.documents.length > 0 ) {
    doc = app.activeDocument;
    numCompoundPaths = doc.layers[0].compoundPathItems.length;
}
```

Dataset

一组用于动态发布的数据。数据集使您可以将多个变量及其动态数据收集到一个对象中。为了创建数据集，您必须至少有一个绑定到图稿的变量。参见类变量。[Variable](#)。

Dataset properties

属性	值类型	注释
name	string	数据集名称
parent	Document	只读。包含此数据集的对象的名称。
typename	string	只读。被引用对象的类名。

Dataset methods

方法	参数类型	返回值	作用
display ()		Nothing	显示数据集。
remove ()		Nothing	删除数据集。
update ()		Nothing	更新数据集。

使用变量和数据集

//创建两个变量，即1个可见性和1个文本，创建两个数据集，每个变量均具有不同的值，然后显示两个数据集

```
var docRef = documents.add();

// 创建可见性变量
var itemRef = docRef.pathItems.rectangle(600, 200, 150, 150);
var colorRef = new RGBColor;
colorRef.red = 255;
itemRef.fillColor = colorRef;
var visibilityVar = docRef.variables.add();
visibilityVar.kind = VariableKind.VISIBILITY;
itemRef.visibilityVariable = visibilityVar;

// 创建文本变量
var textRef = docRef.textFrames.add();
textRef.contents = "Text Variable, dataset 1";
textRef.top = 400;
textRef.left = 200;
var textVar = docRef.variables.add();
textVar.kind = VariableKind.TEXTUAL;
```



```
textRef.contentVariable = textVar;
redraw();

// 创建数据集1
var ds1 = docRef.dataSets.add();

//更改变量值并创建数据集2
itemRef.hidden = true;
textRef.contents = "Text Variable, dataset
2"; redraw();
var ds2 = docRef.dataSets.add();

// 显示每个数据集
ds1.display();
redraw();
ds2.display();
redraw();
```

Datasets

数据集对象的集合。

Datasets properties

属性	值类型	What is it
length	number	只读。集合中的数据集数量
parent	Document	只读。包含此数据集的对象的名称
typename	string	只读。只读。被引用对象的类名。

Datasets methods

方法	参数类型	返回值	作用
add ()		Dataset	创建一个新的数据集对象。
getByName (name)	string	Dataset	获取具有指定名称的集合中的一个元素。
index (itemKey)	string, number	Dataset	从集合中获取一个元素。
removeAll ()		Nothing	删除集合中的所有元素。

Document

Illustrator文档。文档包含在Application对象中。

默认文档设置（以单词“default”开头的那些属性）是影响当前文档的全局设置。确保仅在打开文档时才修改这些默认属性。请注意，如果在创建新对象之前将默认属性设置为所需值，则可以简化脚本，从而无需指定具有默认属性的特定属性（例如fillColor和stroked）。

只能在创建文档时设置文档的颜色空间，高度和宽度。您不能在现有文档中修改这些属性。参见[Application.open](#) 有关如何处理文档色彩空间的更多信息。

Document properties

属性	值类型	注释
activeDataset	Dataset	当前打开的数据集。
activeLayer	Layer	文档中的活动层。
activeView	View	只读。该文档的当前视图。
artboards	Artboards	只读。文档中的所有画板。
brushes	Brushes	只读。文档中包含的画笔。
characterStyles	CharacterStyles	只读。本文档中的字符样式列表。
compoundPathItems	CompoundPathItems	只读。文档中包含的复合路径项。
cropBox	array of 4 numbers	用于输出的文档裁剪框的边界；如果未设置任何值，则为null。
cropStyle	CropOptions	文件裁剪框的样式。
dataSets	Datasets	只读。文档中包含的数据集。
defaultFillColor	Color	如果defaultFilled为true，则用于填充新路径的颜色。
defaultFilled	boolean	如果为true，则应填写新路径。
defaultFillOverprint	boolean	如果为true，则默认情况下应套印填充对象下面的插图。
defaultStrokeCap	StrokeCap	创建的路径的默认行上限类型。
defaultStrokeColor	Color	如果默认笔触为true，则新路径的笔触颜色。
defaultStroked	boolean	如果为true，则应绘制一条新路径。

属性	值类型	注释
<code>defaultStrokeDashes</code>	object	虚线的默认长度和虚线中的间隔，从第一个破折号长度开始，然后是第一个间隔长度，依此类推。设为实线为空对象{}。
<code>defaultStrokeDashOffset</code>	number (double)	虚线模式的默认距离，新路径应从该距离开始。
<code>defaultStrokeJoin</code>	StrokeJoin	新路径中关节的默认类型。
<code>defaultStrokeMiterLimit</code>	number (double)	当默认笔触连接设置为“斜接”时，此属性指定默认情况下何时将连接转换为斜角（平方）。默认的斜接限制为4表示当点的长度达到笔划粗细的四倍时，连接将从斜接连接切换为斜角连接。范围：1到500；值1指定一个斜角连接。
<code>defaultStrokeOverprint</code>	boolean	如果为true，则默认情况下应在描边对象下面的插图上套印。
<code>defaultStrokeWidth</code>	number (double)	新路径的默认笔划宽度。
<code>documentColorSpace</code>	DocumentColorSpace	只读。用于此文档的色彩空间的颜色规范系统。
<code>fullName</code>	File	只读。与文档关联的文件，包括文件的完整路径。
<code>geometricBounds</code>	array of 4 numbers	只读。插图的边界，不包括文档中任何对象的笔触宽度。
<code>gradients</code>	Gradients	只读。文档中包含的渐变。
<code>graphicStyles</code>	GraphicStyles	只读。本文档中定义的图形样式。
<code>graphItems</code>	GraphItems	只读。本文档中的图形图稿项目。
<code>groupItems</code>	GroupItems	只读。文档中包含的群组项目。
<code>height</code>	number (double)	只读。文件的高度。
<code>inkList</code>	object	只读。本文档中的墨水列表。
<code>kinsokuSet</code>	object	只读。Kinsoku字符集，不能以日语文字开头或结尾。

属性	值类型	注释
<code>layers</code>	Layers	只读。文档中包含的图层。
<code>legacyTextItems</code>	LegacyTextItems	只读。文档中的旧文本项。
<code>meshItems</code>	MeshItems	只读。文档中包含的网格图稿。
<code>mojikumiSet</code>	object	只读。预定义的Mojikumi集的名称列表，用于指定日语文本的布局和组成的间距。
<code>name</code>	string	只读。文件名称（不是文件的完整档案路径）。
<code>nonNativeItems</code>	NonNativeItems	只读。本文档中的非本机图稿项目。
<code>outputResolution</code>	number (double)	只读。文档的当前输出分辨率，以每英寸点数（dpi）为单位。
<code>pageItems</code>	PageItems	只读。文档中包含的页面项目（所有图稿项目类）。
<code>pageOrigin</code>	array of 2 numbers	文档中页面的零点（相对于整个高度和宽度），不带边距。
<code>paragraphStyles</code>	ParagraphStyles	只读。本文档中段落样式的列表。
<code>parent</code>	Application	只读。包含此文档的应用程序。
<code>path</code>	File	只读。与文档关联的文件，包括文件的完整路径。
<code>pathItems</code>	PathItems	只读。本文档中包含的路径项。
<code>patterns</code>	Patterns	只读。本文档中包含的模式。
<code>placedItems</code>	PlacedItems	只读。本文档中包含的链接项目。
<code>pluginItems</code>	PluginItems	只读。本文档中包含的插件项。
<code>printTiles</code>	boolean	只读。如果为true，则应将此文档打印为平铺输出。
<code>rasterEffectSettings</code>	RasterEffectOptions	文档的光栅效果设置。

属性	值类型	注释
<code>rasterItems</code>	RasterItems	只读。本文档中包含的栅格项目。
<code>rulerOrigin</code>	array of 2 numbers	文档中标尺相对于文档左下角的零点。
<code>rulerUnits</code>	RulerUnits	只读。文档中标尺的默认度量单位。
<code>saved</code>	boolean	如果为 <code>true</code> ，则自上次保存以来未更改过该文档。
<code>selection</code>	array of objects	对当前文档中所选对象的引用；如果未选择任何内容，则为 <code>null</code> 。 当所选文字图稿项目的内容中存在活动的插入点时，将返回对插入点的引用。类似地，当在文字图稿项目的内容中选择字符时，返回对文本范围的引用。
<code>showPlacedImages</code>	boolean	只读。如果为 <code>true</code> ，则应在文档中显示放置的图像。
<code>splitLongPaths</code>	boolean	只读。如果为 <code>true</code> ，则在打印时应拆分长路径。
<code>spots</code>	Spots	只读。本文档中包含的专色。
<code>stationery</code>	boolean	只读。如果为 <code>true</code> ，则该文件为信纸文件。
<code>stories</code>	Stories	只读。本文档中的段落项目。
<code>swatches</code>	Swatches	只读。本文档中的色板。
<code>swatchGroups</code>	SwatchGroups	只读。本文档中的色板组。
<code>symbolItems</code>	SymbolItems	只读。文档中的符号组。
<code>symbols</code>	Symbols	只读。本文档中的符号。
<code>tags</code>	Tags	只读。本文档中的标签。
<code>textFrames</code>	TextFrameItems	只读。本文档中的文字框架。
<code>tileFullPages</code>	boolean	只读。如果为 <code>true</code> ，则在打印此文档时应平铺整页。
<code>typename</code>	string	只读。被引用对象的类名。

属性	值类型	注释
<code>useDefaultScreen</code>	boolean	只读。如果为true，则在打印此文档时应使用打印机的默认屏幕。
<code>variables</code>	Variables	只读。本文档中定义的变量。
<code>variablesLocked</code>	boolean	如果为true，则变量被锁定。
<code>views</code>	Views	只读。本文档中包含的视图。
<code>visibleBounds</code>	array of 4 numbers	只读。文档的可见范围，包括图中任何对象的笔划宽度。
<code>width</code>	number (double)	只读。该文档的宽度。
<code>XMPString</code>	string	与此文档关联的XMP元数据包。

Document methods

方法	参数类型	返回值	作用
<code>activate</code> ()		Nothing	将与文档关联的第一个窗口置于最前面。
<code>arrange</code> ([layoutStyle])	DocumentLayoutStyle	Boolean	以给定的布局样式排列多个文档。
<code>close</code> ([saveOptions])	SaveOptions	Nothing	使用指定的保存选项关闭文档。 关闭文档时，应将文档引用设置为null，以防止脚本意外尝试访问已关闭的文档。
<code>closeNoUI</code> ()		Nothing	关闭指定的非UI文档。
<code>convertCoordinate</code> (coordinate, source, destination)	Point CoordinateSystem CoordinateSystem	Point	在画板和文档坐标系之间转换给定点。返回值转换后的点坐标。

方法	参数类型	返回值	作用
exportFile (exportFile, exportFormat [,options])	File ExportType variant	Nothing	使用预定义的导出文件格式之一将文档导出到指定文件。除Photoshop®文档外,适当的文件扩展名会自动附加到文件名中。对于这些,您必须在文件规范中包括文件扩展名(PSD)。
exportPDFPreset (file)	File	Nothing	将当前的PDF预设值导出到文件中。
exportPerspectiveGridPreset (file)	File	Nothing	将当前透视图网格预设值导出到文件。
exportPrintPreset (file)	File	Nothing	将当前的打印预设值导出到文件中。
exportVariables (fileSpec)	File	Nothing	将数据集保存到XML库中。数据集包含变量及其关联的动态数据。
fitArtboardToSelectedArt ([index])	number (long)	boolean	按照给定的索引调整画板的大小以适合当前选择的画。索引默认值为0。成功则返回true。
getPerspectiveActivePlane ()		PerspectiveGridPlaneType	检索文档的活动透视图网格的活动平面。
hidePerspectiveGrid ()		boolean	隐藏文档的当前活动网格。如果看不到网格,则不执行任何操作。如果隐藏了网格,则返回true。
imageCapture (imageFile, [clipBounds], [options])	File Rect ImageCaptureOptions	Nothing	在此文档的剪切范围内捕获图稿内容作为光栅图像,并将图像数据写入指定的文件。如果省略bounds参数,则捕获整个图稿。
importCharacterStyles (fileSpec)	File	Nothing	从Illustrator文件中加载字符样式。

方法	参数类型	返回值	作用
importParagraphStyles (fileSpec)	File	Nothing	从Illustrator文件加载段落样式。
importPDFPreset (fileSpec [, replacingPreset])	File boolean	Nothing	从文件加载所有PDF预设。
importPerspectiveGridPreset (fileSpec [, perspectivePreset])	File String	Nothing	加载指定的透视图网格预设，或者，如果未指定预设，则加载文件中的所有预设。
importPrintPreset (printPreset, fileSpec)	string File	Nothing	从文件加载命名的打印预设。
importVariables (fileSpec)	File	Nothing	导入一个包含数据集，变量及其关联的动态数据的库。导入变量将覆盖现有变量和数据集。
print ([options])	PrintOptions	Nothing	打印文档。
rasterize (sourceArt [, clipBounds] [, options])	variant Rect RasterizeOptions	RasterItem	在指定的剪辑范围内栅格化源图稿作品。作为栅格化的结果，丢弃了源图稿。
rearrangeArboards ([artboardLayout, artboardRowsOrCols, artboardSpacing, artboardMoveArtwork])	DocumentArtboardLayout integer Number boolean	boolean	重新排列文档中的画板。所有参数都是可选的。默认布局样式为 DocumentArtboardLayout.GridByRow 。 第二个参数在[1..docNumArtboards-1]范围内指定适合所选布局样式的行数或列数，对于单行/列布局则指定1（默认值）。 间距是像素数，默认为20。 当最后一个参数为true（默认值）时，图稿将随画板一起移动。

方法	参数类型	返回值	作用
save ()		Nothing	将文档保存在当前位置。
saveAs (saveIn [, options])	File SaveOptions	Nothing	将文档另存为指定文件中的Illustrator, EPS或PDF文件。
saveNoUI (saveIn)	File	Nothing	将非UI文档保存在指定路径
selectObjectsOnActiveArtboard ()		boolean	选择当前活动画板上的对象。成功返回true。
setActivePlane (gridPlane)	PerspectiveGridPlaneType	boolean	设置文档的活动透视图网格的活动平面。成功返回true。
selectPerspectivePreset (gridType, presetName)	PerspectiveGridType string	boolean	选择预定义的预设当前文档定义网格。成功返回true。
showPerspectiveGrid ()		boolean	显示文档的当前活动网格, 或者如果没有活动的网格, 则显示默认网格。成功返回true。
windowCapture (imageFile, windowSize)	File array of 2 numbers	Nothing	将当前文档窗口捕获到目标TIFF图像文件。

取消选择当前文档中的所有对象

可以将最前面的文档称为activeDocument或documents [0]。

```
var docRef = activeDocument;
docRef.selection = null;
```

关闭文件

//关闭活动文档而不保存更改

```
if ( app.documents.length > 0 ) {

    aiDocument = app.activeDocument;
    aiDocument.close( SaveOptions.DONOTSAVECHANGES );
    aiDocument = null;
}
```

创建具有默认值的文档

```
//如果不存在则创建一个新文档
//然后将填充和笔触默认设置为true

if ( app.documents.length == 0 ) {
    doc = app.documents.add();
}
else {
    doc = app.activeDocument;
}
doc.defaultFilled = true;
doc.defaultStroked = true;
```

DocumentPreset

创建新文档时要使用的预设文档模板。参见 `Documents.addDocument()`

DocumentPreset properties

属性	值类型	注释
<code>artboardLayout</code>	DocumentArtboardLayout	新文档中画板的布局。默认值： GridByRow
<code>artboardRowsOrCols</code>	number (long)	画板的行数（对于行布局）或列数（对于列布局）。范围：1到 (<code>numArtboards-1</code>) 或1（单行或列布局）。默认值：1
<code>artboardSpacing</code>	number (double)	新文档中画板之间的间距。默认值：20.0
<code>colorMode</code>	DocumentColorSpace	新文档的色彩空间。
<code>documentBleedLink</code>	Boolean	出血值的文档链接。
<code>documentBleedOffsetRectangle</code>	rectangle	文档出血偏移矩形。
<code>height</code>	number (double)	文档中的高度（以磅为单位）。默认值：792.0
<code>numArtboards</code>	number (long)	新文档的画板数。范围：1到100。 默认值：1
<code>previewMode</code>	DocumentPreviewMode	新文档的预览模式。
<code>rasterResolution</code>	DocumentRasterResolution	新文档的栅格分辨率。
<code>title</code>	string	文件标题。
<code>transparencyGrid</code>	DocumentTransparencyGrid	新文档的透明网格颜色。
<code>typename</code>	string	只读。被引用对象的类名。
<code>units</code>	RulerUnits	新文档的标尺单位。
<code>width</code>	number (double)	文档点的宽度。默认值： 612.0

Documents

Document对象的集合。

Documents properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

Documents methods

方法	参数类型	返回值	作用
add ([documentColorSpace] [, width] [, height] [, numArtBoards] [, artboardLayout] [, artboardSpacing] [, artboardRowsOrCols])	DocumentColorSpace number (double) number (double) number (long) DocumentArtboardLayout number (double) number (long)	Document	创建一个新文件 使用可选参数并返回对新文档的引用。
addDocument (startupPreset [, presetSettings] [, showOptionsDialog])	string DocumentPreset boolean	Document	建立文件 从预设中替换所有提供的设置值，并返回对新文档的引用。
addDocumentNoUI (startupPreset)	string	Document	建立文件 不在用户界面中显示
getByName (name)	string	Document	获取指定名称的文档。
index (itemKey)	string, number	Document	从集合中获取一个元素。

创建一个新文件

```
// 用RGB颜色空间创建一个新文档
app.documents.add( DocumentColorSpace.RGB );
```

EPSSaveOptions

用于将文档另存为Illustrator EPS文件的选项，与[saveAs](#)方法一起使用。所有属性都是可选的。

EPSSaveOptions properties

属性	值类型	注释
<code>artboardRange</code>	string	如果 <code>saveMultipleArtboards</code> 为true，则考虑进行多资产提取，以指定画板范围。空字符串将提取所有画板。默认值：空字符串
<code>cmykPostScript</code>	boolean	如果为true，请使用CMYK PostScript。
<code>compatibility</code>	Compatibility	指定要保存的EPS文件格式的版本。默认值： Compatibility.ILLUSTRATOR1719
<code>compatibleGradientPrinting</code>	boolean	如果为true，则创建渐变或渐变网格的栅格项，以便PostScript Level 2打印机可以打印对象。默认值：false
<code>embedAllFonts</code>	boolean	如果为true，则文档使用的所有字体都应嵌入到保存的文件中（版本7或更高版本）。默认值：false
<code>embedLinkedFiles</code>	boolean	如果为true，则链接的图像文件将包含在保存的文档中。
<code>flattenOutput</code>	OutputFlattening	对于早于Illustrator 9的文件格式，透明度应如何调整？
<code>includeDocumentThumbnails</code>	boolean	如果为true，则应包括EPS图稿的缩略图。
<code>overprint</code>	PDFOverprint	是保留，丢弃还是模拟叠印。 默认： PDFOverprint.PRESERVEPDFOVERPRINT
<code>postScript</code>	EPSPostScriptLevelEnum	要使用的PostScript语言级别（级别1对文件格式版本8或更早版本有效）。默认： EPSPostScriptLevelEnum.LEVEL2
<code>preview</code>	EPSPreview	EPS预览图像的格式。
<code>saveMultipleArtboards</code>	boolean	如果为true，则保存所有画板或画板范围。默认值：false
<code>typename</code>	string	只读。被引用对象的类名。

Exporting to EPS format

```
//将当前文档导出为destFile作为具有指定选项的EPS文件，  
// destFile包含完整路径，包括文件名  
function exportFileAsEPS (destFile) {  
  
    var newFile = new File(destFile);  
    var saveDoc;  
    if ( app.documents.length == 0 )  
        saveDoc = app.documents.add();  
    else  
        saveDoc = app.activeDocument;  
    var saveOpts = new EPSSaveOptions();  
    saveOpts.cmykPostScript = true;  
    saveOpts.embedAllFonts = true;  
    saveDoc.saveAs( newFile, saveOpts );  
}
```

ExportOptionsAutoCAD

用于将文档导出为AutoCAD文件的选项，与 [exportFile](#) 方法一起使用。所有属性都是可选的。

导出文档时，文件扩展名会自动添加。您不应在文件规范中包括任何文件扩展名。要覆盖默认的AutoCAD导出格式（DWG），请使用 [exportFileFormat](#) 属性。

ExportOptionsAutoCAD properties

属性	值类型	注释
<code>alterPathsForAppearance</code>	boolean	如果为true，则需要更改路径以保持外观。默认值: false
<code>colors</code>	AutoCADColors	颜色已导出到AutoCAD文件中。
<code>convertTextToOutlines</code>	boolean	如果为true，则将文本转换为矢量路径；否则为false。保留字体的外观。默认值: false
<code>exportFileFormat</code>	AutoCADExportFileFormat	文件导出到的格式。默认: AutoCADExportFileFormat.DWG
<code>exportOption</code>	AutoCADExportOption	指定在导出过程中保留外观还是可编辑性。默认: AutoCADExportOption. MaximizeEditability
<code>exportSelectedArtOnly</code>	boolean	如果为true，则仅导出所选图稿。默认值: false
<code>rasterFormat</code>	AutoCADRasterFormat	导出光栅图样的格式。
<code>scaleLineweights</code>	boolean	如果为true，则按与图形其余部分相同的比例因子缩放线宽。默认值: false
<code>typename</code>	string	只读。被引用对象的类名。
<code>unit</code>	AutoCADUnit	要映射的度量单位。

属性	值类型	注释
<code>unitScaleRatio</code>	number (double)	缩放比例的比例 (百分比)。范围: 0至1000
<code>version</code>	AutoCADCompatibility	将文件导出到的AutoCAD版本。默认: AutoCADCompatibility.AutoCADRelease24

ExportOptionsFlash

用于将文档导出为Macromedia®Flash®（SWF）文件的选项，与 [exportFile](#) 方法一起使用。所有属性都是可选的。

导出文档时，将自动附加适当的文件扩展名。您不应在文件规范中包括任何文件扩展名。

ExportOptionsFlash properties

属性	值类型	注释
artClipping	ArtClippingOption	在输出过程中应如何裁剪图稿作品。默认： ArtClippingOption.OUTPUTARTBOUNDS
artboardRange	string	如果saveMultipleArtboards为true，则考虑进行多资产提取，以指定画板范围。空字符串将提取所有画板。默认：empty string
backgroundColor	RGBColor	导出的Flash框架的背景色。
backgroundLayers	array of Layer	包含的层列表，作为导出的Flash框架的静态背景。
blendAnimation	BlendAnimationType	混合对象的动画类型。默认： BlendAnimationType.NOBLENDANIMATION
compressed	boolean	如果为true，则应以压缩方式导出导出的文件。 默认值：false
convertTextToOutlines	boolean	如果为true，则所有文本都将转换为矢量路径；保留所有Flash Player中字体的视觉外观。默认值：false
curveQuality	number (long)	应显示的曲线信息量。默认值：7
exportAllSymbols	boolean	如果为true，则导出调色板中定义的所有符号。默认值：false
exportStyle	FlashExportStyle	在Flash中创建导出数据样式。 默认：FlashExportStyle.ASFLASHFILE
exportVersion	FlashExportVersion	导出的SWF文件的版本。 默认：FlashExportVersion.FlashVersion9.
frameRate	number (double)	显示速率，以每秒帧数为单位。范围：0.01–120.0。默认值：12.0
ignoreTextKerning	boolean	如果为true，则忽略文本对象中的字距调整信息。默认值：false

属性	值类型	注释
<code>imageFormat</code>	FlashImageFormat	如何压缩导出的Flash文件中的图像。 默认: <code>FlashImageFormat.LOSSLESS</code>
<code>includeMetadata</code>	boolean	如果为true, 则在SWF文件中包括最少的XMP元数据。默认: <code>false</code>
<code>jpegMethod</code>	FlashJPEGMethod	指定要使用的JPEG方法。默认: <code>FlashJPEGMethod.Standard</code>
<code>jpegQuality</code>	number (long)	使用的压缩级别。范围1到10。默认值: 3
<code>layerOrder</code>	LayerOrderType	图层导出到Flash帧的顺序。默认: <code>LayerOrderType.BOTTOMUP</code>
<code>looping</code>	boolean	如果为true, 则将Flash文件设置为在运行时循环播放。默认值: <code>false</code>
<code>playbackAccess</code>	FlashPlaybackSecurity	导出的SWF文件的访问级别。 默认: <code>FlashPlaybackSecurity.PlaybackLocal</code>
<code>preserveAppearance</code>	boolean	如果为true, 请保留外观。如果为false, 则保留可编辑性。默认值: <code>false</code>
<code>readOnly</code>	boolean	如果为true, 则导出为只读文件。默认值: <code>false</code>
<code>replacing</code>	SaveOptions	如果已经存在同名文件, 则应将其替换。 默认: <code>SaveOptions.PROMPTTOSAVECHANGES</code>
<code>resolution</code>	number (double)	分辨率, 以每英寸像素为单位。范围: 72–2400。默认值: 72
<code>saveMultipleArtboards</code>	boolean	如果为true, 则保存所有画板或画板范围。默认值: <code>false</code>
<code>typename</code>	string	只读。被引用对象的类名。

导出为Flash格式

//将当前文档作为具有指定选项的Flash文件导出到destFile, // destFile包含完整路径, 包括文件名

```
function exportToFlashFile(destFile) {
    if ( app.documents.length > 0 ) {
        var exportOptions = new ExportOptionsFlash();
        var type = ExportType.FLASH;
        var fileSpec = new File(destFile);
        exportOptions.resolution = 150;
        app.activeDocument.exportFile( fileSpec, type, exportOptions );
    }
}
```

ExportOptionsGIF

用于将文档导出为GIF文件的选项，与 [exportFile](#) 方法一起使用。所有属性都是可选的。

导出文档时，将自动附加适当的文件扩展名。您不应在文件规范中包括任何文件扩展名。

ExportOptionsGIF properties

属性	值类型	注释
antiAliasing	boolean	如果为true，则应对导出的图像进行抗锯齿处理。默认值：true
artBoardClipping	boolean	如果为true，则应将导出的图像剪切到画板上。默认值：false
colorCount	number (long)	导出图像的颜色表中的颜色数。范围：2到256。默认值：128
colorDither	ColorDitherMethod	用于抖动导出图像中颜色的方法。 默认：ColorDitherMethod.DIFFUSION
colorReduction	ColorReductionMethod	用于减少导出图像中颜色数量的方法。 默认：ColorReductionMethod.SELECTIVE
ditherPercent	number (long)	应该对输出图像的颜色进行多少抖动处理，其中100.0是100%。
horizontalScale	number (double)	要应用于导出图像的水平比例因子，其中100.0是100%。默认值：100.0
infoLossPercent	number (long)	压缩期间允许的信息丢失级别，其中100.0为100%。
interlaced	boolean	如果为true，则应该对导出的图像进行隔行扫描。默认值：false
matte	boolean	如果为true，则画板应带有颜色。默认值：true
matteColor	RGBColor	填充画板时要使用的颜色。默认值：白色
saveAsHTML	boolean	如果为true，则导出的图像应与随附的HTML文件一起保存。默认值：false
transparency	boolean	如果为true，则导出的图像应使用透明度。默认值：true
typename	string	只读。被引用对象的类名。

属性	值类型	注释
<code>verticalScale</code>	number (double)	应用于输出图像的垂直比例因子，其中100.0是100%。默认值：100.0
<code>webSnap</code>	number (long)	更改颜色表以匹配Web调色板的数量（最大为100）。默认值：0

导出为GIF格式

//将当前文档作为具有指定选项的GIF文件导出到dest,
// dest包含完整路径，包括文件名

```
function exportToGIFFile(dest) {  
    if ( app.documents.length > 0 ) {  
        var exportOptions = new ExportOptionsGIF();  
        var type = ExportType.GIF;  
        var fileSpec = new File(dest);  
  
        exportOptions.antiAliasing = false;  
        exportOptions.colorCount = 64;  
        exportOptions.colorDither = ColorDitherMethod.DIFFUSION;  
  
        app.activeDocument.exportFile( fileSpec, type, exportOptions );  
    }  
}
```

ExportOptionsJPEG

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

导出文档时，将自动附加适当的文件扩展名。您不应在文件规范中包括任何文件扩展名。

ExportOptionsJPEG properties

属性	值类型	注释
antiAliasing	boolean	如果为true，则应对导出的图像进行抗锯齿处理。默认值: true
artBoardClipping	boolean	如果为true，则应将导出的图像剪切到画板上。
blurAmount	number (double)	应用于导出图像的模糊量。范围: 0.0到2.0。默认值: 0.0
horizontalScale	number (double)	要应用于导出图像的水平比例因子，其中100.0是100%。默认值: 100.0
matte	boolean	如果为true，则画板上应带有某种颜色。默认值: true
matteColor	RGBColor	填充画板时要使用的颜色。默认值: 白色
optimization	boolean	如果为true，则应优化导出的图像以进行web查看。默认值: true
qualitySetting	number (long)	导出图像的质量。范围: 0到100。默认值: 30
saveAsHTML	boolean	如果为true，则导出的图像应与随附的HTML文件一起保存。默认值: false
typename	string	只读。被引用对象的类名。
verticalScale	number (double)	应用于输出图像的垂直比例因子。范围: 0.0至776.19。默认值: 100.0

导出为JPEG格式

//使用指定的选项将当前文档导出为dest作为JPEG文件，dest包含完整路径，包括文件名

```
function exportFileToJPEG (dest) {  
    if ( app.documents.length > 0 ) {  
        var exportOptions = new ExportOptionsJPEG();  
        var type = ExportType.JPEG;  
        var fileSpec = new File(dest);  
        exportOptions.antiAliasing = false;  
        exportOptions.qualitySetting = 70;  
        app.activeDocument.exportFile( fileSpec, type, exportOptions );  
    }  
}
```

ExportOptionsPhotoshop

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

导出文档时，将自动附加适当的文件扩展名。您不应在文件规范中包括任何文件扩展名。

ExportOptionsPhotoshop properties

属性	值类型	注释
antiAliasing	boolean	如果为true，则应对导出的图像进行抗锯齿处理。默认值: true
artboardRange	string	如果saveMultipleArtboards为true，则考虑进行多资产提取，以指定画板范围。空字符串将提取所有画板。默认值: 空字符串
editableText	boolean	如果为true，则应将文本对象导出为可编辑文本层。默认值: true
embedICCProfile	boolean	如果为true，则应在导出的文件中嵌入ICC配置文件。默认值: false
imageColorSpace	ImageColorSpace	导出文件的色彩空间。 默认: ImageColorSpace.RGB
maximumEditability	boolean	导出时，请保留尽可能多的原始文档结构。默认值: true
resolution	number (double)	导出文件的分辨率，以每英寸点数 (dpi) 为单位。范围: 72.0至2400.0。默认值: 150.0
saveMultipleArtboards	boolean	如果为true，则保存所有画板或画板范围。默认值: false
typename	string	只读。被引用对象的类名。
warnings	boolean	如果为true，则在导出设置冲突的情况下应显示警告对话框。默认值: true
writeLayers	boolean	如果为true，则应在导出的文档中显示文档层。默认值: true

导出为Photoshop格式

//使用指定的选项将当前文档作为PSD文件导出到dest,
// dest包含完整路径, 包括文件名

```
function exportFileToPSD (dest) {  
    if ( app.documents.length > 0 ) {  
        var exportOptions = new ExportOptionsPhotoshop();  
        var type = ExportType.PHOTOSHOP;  
        var fileSpec = new File(dest);  
        exportOptions.resolution = 150;  
        app.activeDocument.exportFile( fileSpec, type, exportOptions );  
    }  
}
```

ExportOptionsPNG8

Options for exporting a document as an 8-bit PNG file, used with the [exportFile](#) method. All properties are optional.

导出文档时，将自动附加适当的文件扩展名。您不应在文件规范中包括任何文件扩展名。

ExportOptionsPNG8 properties

属性	值类型	注释
antiAliasing	boolean	如果为true，则应对导出的图像进行抗锯齿处理。默认值: true
artBoardClipping	boolean	如果为true，则应将导出的图像剪切到画板上。默认值: false
colorCount	number (long)	导出图像的颜色表中的颜色数。范围: 2到256。默认值: 128
colorDither	ColorDitherMethod	用于抖动导出图像中颜色的方法。默认: ColorDitherMethod.Diffusion
colorReduction	ColorReductionMethod	用于减少导出图像中颜色数量的方法。默认: ColorReductionMethod.SELECTIVE
ditherPercent	number (long)	所导出图像的颜色抖动的数量 (百分比)，其中100.0是100%。范围: 0到100。默认值: 88
horizontalScale	number (double)	要应用于导出图像的水平比例因子，其中100.0是100%。默认值: 100.0
interlaced	boolean	如果为true，则应该对导出的图像进行隔行扫描。默认值: false
matte	boolean	如果为true，则美工板应带有颜色。默认值: true
matteColor	RGBColor	填充画板时要使用的颜色。默认值: 白色
saveAsHTML	boolean	如果为true，则将导出的图像与随附的HTML文件一起保存。默认值: false
transparency	boolean	如果为true，则导出的图像使用透明度。默认值: true
typename	string	只读。被引用对象的类名。

属性	值类型	注释
<code>verticalScale</code>	number (double)	应用于输出图像的垂直缩放因子，其中100.0为100。默认值：100.0
<code>webSnap</code>	number (long)	指定应更改颜色表以匹配Web调色板的数量，最大值为100。默认值：0

导出为PNG8格式

//使用指定的选项将当前文档作为PNG8文件导出到dest，
// dest包含完整路径，包括文件名

```
function exportFileToPNG8 (dest) {  
    if ( app.documents.length > 0 ) {  
        var exportOptions = new ExportOptionsPNG8();  
        var type = ExportType.PNG8;  
        var fileSpec = new File(dest);  
        exportOptions.colorCount = 8;  
        exportOptions.transparency = false;  
        app.activeDocument.exportFile( fileSpec, type, exportOptions );  
    }  
}
```

ExportOptionsPNG24

用于将文档导出为SVG文件的选项，与 [exportFile](#) 方法一起使用。所有属性都是可选的。

导出文档时，将自动附加适当的文件扩展名。您不应在文件规范中包括任何文件扩展名。

ExportOptionsPNG24 properties

属性	值类型	注释
antiAliasing	boolean	如果为true，则对导出的图像进行抗锯齿处理。默认值: true
artBoardClipping	boolean	如果为true，则将导出的图像剪切到画板上。默认值: false
horizontalScale	number (double)	要应用于导出图像的水平比例因子，其中100.0是100%。默认值: 100.0
matte	boolean	如果为true，则美工板上将带有颜色。默认值: true
matteColor	RGBColor	填充美术板时要使用的颜色。默认值: 白色
saveAsHTML	boolean	如果为true，则将导出的图像与随附的HTML文件一起保存。默认值: false
transparency	boolean	如果为true，则导出的图像使用透明度。默认值: true
typename	string	只读。被引用对象的类名。
verticalScale	number (double)	应用于输出图像的垂直缩放因子，其中100.0为100。默认值: 100.0

导出为PNG24格式

//使用指定的选项将当前文档导出为dest为PNG24文件，// dest包含包含文件名的完整路径，saveAsHTML //选项使用images文件夹中的PNG文件创建HTML版本

```
function exportFileToPNG24 (dest) {
    if ( app.documents.length > 0 ) {
        var exportOptions = new ExportOptionsPNG24();
        var type = ExportType.PNG24;
        var fileSpec = new File(dest);
        exportOptions.antiAliasing = false;
        exportOptions.transparency = false;
        exportOptions.saveAsHTML = true;
        app.activeDocument.exportFile( fileSpec, type, exportOptions );
    }
}
```

ExportOptionsSVG

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

导出文档时，将自动附加适当的文件扩展名。您不应在文件规范中包括任何文件扩展名。

ExportOptionsSVG properties

属性	值类型	注释
artboardRange	string	如果saveMultipleArtboards为true，则要保存的一系列画板。以英文逗号分隔的画板名称列表，或用于保存所有画板的空字符串。默认值：空字符串
compressed	boolean	元素坐标值的小数精度。范围：1到7。默认值：3
coordinatePrecision	number (long)	元素坐标值的小数精度。范围：1到7。默认值：3
cssProperties	SVGCSSTypeLocation	如何将文档的CSS属性包含在导出的文件中。默认： SVGCSSTypeLocation. STYLEATTRIBUTES
documentEncoding	SVGDocumentEncoding	文档中的文本应如何编码。默认： SVGDocumentEncoding.ASCII
DTD	SVGDTDVersion	文件应符合的SVG版本。默认： SVGDTDVersion.SVG1_1
embedRasterImages	boolean	如果为true，则应将文档中包含的光栅图像嵌入导出的文件中。默认值：false
fontSubsetting	SVGFontSubsetting	哪些字体字形应包含在导出的文件中。默认： SVGFontSubsetting.ALLGLYPHS
fontType	SVGFontType	导出文件中包含的字体类型。默认：SVGFontType.CEFFONT
includeFileInfo	boolean	如果为true，则文件信息应保存在导出的文件中。默认值：false

属性	值类型	注释
<code>includeUnusedStyles</code>	boolean	如果为true, 则将未使用的样式保存在导出的文件中。默认值: false
<code>includeVariablesAndDatasets</code>	boolean	如果为true, 则变量和数据集应保存在导出的文件中。默认值: false
<code>optimizeForSVGViewer</code>	boolean	如果为true, 则应针对SVG Viewer 优化导出的文件。默认值: false
<code>preserveEditability</code>	boolean	如果为true, 则在导出文档时应保留Illustrator编辑功能。默认值: false
<code>saveMultipleArtboards</code>	boolean	如果为true, 则将artboardRange指定的画板保存在导出的文件中。默认值: false
<code>slices</code>	boolean	如果为true, 则切片数据应与文件一起导出。默认值: false
<code>SVGAutoKerning</code>	boolean	如果为true, 则文件中允许SVG自动字距调整。默认值: false
<code>SVGTextOnPath</code>	boolean	如果为true, 则文件中允许使用SVG路径上的文本构造。默认值: false
<code>typename</code>	string	只读。被引用对象的类名。

导出为SVG格式

//使用指定的选项将当前文档导出为dest作为SVG文件, // dest包含完整路径, 包括文件名

```
function exportFileToSVG (dest) {
  if ( app.documents.length > 0 ) {
    var exportOptions = new ExportOptionsSVG();
    var type = ExportType.SVG;
    var fileSpec = new File(dest);
    exportOptions.embedRasterImages = true;
    exportOptions.embedAllFonts = false;
    exportOptions.fontSubsetting = SVGFontSubsetting.GLYPHSUSED;
    app.activeDocument.exportFile( fileSpec, type, exportOptions );
  }
}
```

ExportOptionsTIFF

用于将文档导出为TIFF文件的选项，与 [exportFile](#) 方法一起使用。所有属性都是可选的。

导出文档时，将自动附加适当的文件扩展名。您不应在文件规范中包括任何文件扩展名。

ExportOptionsTIFF properties

属性	值类型	注释
antiAliasing	boolean	如果为true，则应对导出的图像进行抗锯齿处理。默认值：true
artboardRange	string	如果saveMultipleArtboards为true，则考虑进行多资产提取，以指定画板范围。空字符串将提取所有画板。默认值：空字符串
byteOrder	TIFFByteOrder	在新文件中使用的字节顺序。
imageColorSpace	ImageColorSpace	导出文件的色彩空间。 默认：ImageColorSpace.RGB
IZWCompression	boolean	如果为true，请在新文件中使用IZW压缩。
resolution	number (double)	导出文件的分辨率，以每英寸点数 (dpi) 为单位。范围：72.0至2400.0。默认值：150.0
saveMultipleArtboards	boolean	如果为true，则保存所有画板或画板范围。默认值：false

导出为TIFF格式

```
//使用指定的选项将当前文档作为TIFF文件导出到dest，
// dest包含完整路径，包括文件名
```

```
function exportFileToPSD (dest) {
  if ( app.documents.length > 0 ) {
    var exportOptions = new ExportOptionsTIFF();
    var type = ExportType.TIFF;
    var fileSpec = new File(dest);
    exportOptions.resolution = 150;
    exportOptions.byteOrder = TIFFByteOrder.IBMPC;
    exportOptions.IZWCompression = false;
    app.activeDocument.exportFile( fileSpec, type, exportOptions );
  }
}
```

FXGSaveOptions

指定将文档另存为FXG文件时可能提供的选项。所有属性都是可选的。

FXGSaveOptions properties

属性	值类型	注释
<code>artboardRange</code>	string	如果 <code>saveMultipleArtboards</code> 为true，则考虑进行多资产提取，以指定画板范围。空字符串将提取所有画板。 默认值：空字符串
<code>blendsPolicy</code>	BlendsExpandPolicy	FXG用来扩展混合的策略。默认值： <code>BlendsExpandPolicy.AUTOMATICALLYCONVERTBLEND</code>
<code>downsampleLinkedImages</code>	boolean	如果为true，则对链接的图像进行降采样（72 dpi）。默认值： <code>false</code>
<code>filtersPolicy</code>	FiltersPreservePolicy	FXG用于保留过滤器的策略。默认值： <code>FiltersPreservePolicy</code> 。保持可编辑
<code>gradientsPolicy</code>	GradientsPreservePolicy	FXG用于保留渐变的策略。默认： <code>GradientsPreservePolicy.AUTOMATICALLYCONVERTGRADIENTS</code>
<code>includeUnusedSymbols</code>	boolean	如果为true，则包括未使用的符号。默认值： <code>false</code>
<code>preserveEditingCapabilities</code>	boolean	如果为true，则保留FXG的编辑功能。默认值： <code>true</code>
<code>saveMultipleArtboards</code>	boolean	如果为true，则保存所有画板或画板范围。默认值： <code>false</code>
<code>textPolicy</code>	TextPreservePolicy	FXG用于保留文本的策略。默认： <code>TextPreservePolicy.AUTOMATICALLYCONVERTTEXT</code>
<code>version</code>	FXGVersion	要创建的FXG文件格式的版本。默认 <code>FXGVersion.VERSION2PT0</code>

Gradient

文档中包含的渐变定义。脚本可以创建新的渐变。

Gradient properties

属性	值类型	注释
<code>gradientStops</code>	GradientStops	只读。渐变停止包含在此渐变中。
<code>name</code>	string	渐变的名称。
<code>parent</code>	Document	只读。包含此渐变的文档。
<code>type</code>	GradientType	渐变的种类，可以是径向的或线性的。
<code>typename</code>	string	只读。被引用对象的类名。

Gradient methods

方法	参数类型	返回值	作用
<code>remove()</code>		Nothing	从文档中删除引用的对象。

创建和应用渐变

```
//在当前文档中创建新的渐变，然后
//将渐变应用于最前面的路径项

if ( app.documents.length > 0 ) {
    //为渐变的两端创建颜色
    var startColor = new RGBColor();
    var endColor = new RGBColor();

    startColor.red = 0;
    startColor.green = 100;
    startColor.blue = 255;
    endColor.red = 220;
    endColor.green = 0;
    endColor.blue = 100;

    //创建一个新的渐变
    //一个新的渐变总是有2个停靠点
    var newGradient = app.activeDocument.gradients.add();
    newGradient.name = "NewGradient";
    newGradient.type = GradientType.LINEAR;

    //修改第一个渐变停止
    newGradient.gradientStops[0].rampPoint = 30;
    newGradient.gradientStops[0].midPoint = 60;
    newGradient.gradientStops[0].color = startColor;
}
```

```
//修改最后一个渐变停止
newGradient.gradientStops[1].rampPoint = 80;
newGradient.gradientStops[1].color = endColor;

//构造一个
//新创建的渐变的Illustrator.GradientColor对象
var colorOfGradient = new GradientColor();
colorOfGradient.gradient = newGradient;

//获取第一个路径项，应用新的渐变作为其填充
var topPath = app.activeDocument.pathItems[0];
topPath.filled = true;
topPath.fillColor = colorOfGradient;
}
```

GradientColor

渐变对象中的渐变颜色规范。脚本可以使用对文档中现有渐变的引用来创建新的渐变颜色。如果没有引用现有的渐变对象，则提供默认渐变。

GradientColor properties

属性	值类型	注释
angle	number (double)	梯度矢量角度（以度为单位）。默认值：0.0
gradient	Gradient	引用定义渐变的对象。
hiliteAngle	number (double)	渐变高光矢量角度（以度为单位）。
hiliteLength	number (double)	渐变高光矢量长度。
length	number (double)	渐变向量的长度。
matrix	Matrix	附加的变换矩阵可操纵渐变路径。
origin	array of 2 numbers	渐变矢量的原点，即此颜色的渐变中心。
typename	string	只读。被引用对象的类名。

更改渐变停止色

// 创建一个新的RGB文档，然后更改索引渐变的第一个渐变停止的颜色

```
app.documents.add(DocumentColorSpace.RGB);
```

// 获取要更改的渐变的引用

```
var gradientRef = app.activeDocument.gradients[1];
```

// 创建新颜色

```
var startColor = new RGBColor();
```

```
startColor.red = 255;
```

```
startColor.green = 238;
```

```
startColor.blue = 98;
```

// 将新颜色应用于第一个渐变色停止点

```
gradientRef.gradientStops[0].color = startColor;
```

Gradients

文档中渐变对象的集合。

Gradients properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

Gradients methods

方法	参数类型	返回值	作用
add ()		Gradient	创建一个新的Gradient对象。
getByName (name)	string	Gradient	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	Gradient	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

移除渐变

```
//从当前文档中删除第一个渐变
if ( app.documents.length > 0 ) {
    app.activeDocument.gradients[0].remove();
}
```

GradientStop

渐变停止定义，表示文档中定义的特定渐变上的点。每个渐变色标指定包含的渐变中的颜色变化。See [Changing a gradient stop color](#) for an example.

GradientStop properties

属性	值类型	注释
<code>color</code>	Color	与此渐变色标链接的颜色。
<code>midPoint</code>	number (double)	中点键值，指定为13.0到87.0之间的百分比。
<code>opacity</code>	number (double)	渐变色标的不透明度值。范围：0.0至100.0
<code>parent</code>	Gradient	只读。包含此梯度停止点的梯度。
<code>rampPoint</code>	number (double)	颜色在混合物中的位置为0.0到100.0，其中100.0为100%。
<code>typename</code>	string	只读。被引用对象的类名。

GradientStop methods

方法	参数类型	返回值	作用
<code>remove</code> ()		Nothing	删除该对象。

GradientStops

特定渐变中GradientStop对象的集合。元素未命名；您必须按索引访问它们。

GradientStops properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

GradientStops methods

方法	参数类型	返回值	作用
add ()		GradientStop	创建一个新对象。
getName (name)	string	GradientStop	获取具有指定名称的集合中的第一个元素。
index (itemKey)	number	GradientStop	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有对象。

添加新的渐变色标

//向渐变添加新的渐变色标，新色标的颜色为70%灰色

```
if ( app.documents.length > 0 && app.activeDocument.gradients.length > 0 ) {
  //获取要更改的渐变的参考
  var changeGradient = app.activeDocument.gradients[0];
  //获取对最后一个渐变停止点的引用
  var origCount = changeGradient.gradientStops.length;
  var lastStop = changeGradient.gradientStops[origCount-1];
  //添加新的渐变色标
  var newStop = changeGradient.gradientStops.add();

  //设置新的渐变色标的值。
  //将原始的最后-一个渐变色停止点向左移动，
  //在旧位置插入新的渐变色停止点
  newStop.rampPoint = lastStop.rampPoint;
  lastStop.rampPoint = lastStop.rampPoint - 10;
  //创建新的颜色以应用到新创建的渐变色停止
  //-灰色色调值为70%
  var newStopColor = new GrayColor();
  newStopColor.gray = 70.0;
  newStop.color = newStopColor;
}
```

```
}
```

GraphicStyle

图形样式。每种图形样式都定义了一组外观属性，您可以将它们无损地应用于页面项目。图形样式包含在文档中。脚本无法创建新的图形样式。

GraphicStyle properties

属性	值类型	注释
name	string	图形样式名称。
parent	Document	只读。包含此图形样式的文档。
typename	string	只读。被引用对象的类名。

GraphicStyle methods

方法	参数类型	返回值	作用
applyTo (artItem)	PageItem	Nothing	将此图形样式应用于指定的美术项目。
mergeTo (artItem)	PageItem	Nothing	将此图形样式合并为指定美术项目的当前样式。
remove ()		Nothing	删除该对象。

应用图形样式

```
//复制选择中的每个路径项，将复制项放到
//新的组中，然后将图形样式应用于新的组项
```

```
if ( app.documents.length > 0 ) {
var doc = app.activeDocument;
var selected = doc.selection;

var newGroup = doc.groupItems.add();
newGroup.name = "NewGroup";
newGroup.move( doc, ElementPlacement.PLACEATEND );

var endIndex = selected.length;
for ( i = 0; i < endIndex; i++ ) {
    if ( selected[i].typename == "PathItem" )
        selected[i].duplicate( newGroup, ElementPlacement.PLACEATEND );
}
for ( i = 0; i < newGroup.pageItems.length; i++ ) {
    doc.graphicStyles[1].applyTo( newGroup.pageItems[i] );
}
}
```


GraphicStyles

文档中GraphicStyle对象的集合。

GraphicStyles properties

属性	值类型	注释
length	number	只读。文档中图形样式的数量。
parent	object	只读。包含此图形样式集合的文档。
typename	string	只读。被引用对象的类名。

GraphicStyles methods

方法	参数类型:	返回值	作用
getByName (name)	string	GroupItem	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	GroupItem	从集合中获取一个元素。
removeAll ()		Nothing	删除引用的集合中的所有元素。

计算图形样式

```
//计算活动文档中图形样式的数量//并将结果存储在numberOfStyles中
if ( app.documents.length > 0 ) {
    var numberOfStyles = app.activeDocument.graphicStyles.length;
}
```

GraphItem

任何图形图稿对象。请参阅下面的示例 [Rotating graph items](#) 旋转图项目。

GraphItem properties

属性	值类型	注释
artworkKnockout	KnockoutState	此对象用于创建淘汰对象吗？如果是，则使用哪种淘汰对象。您不能将此值设置为 <code>KnockoutState.Unknown</code>
blendingMode	BlendModes	合成对象时使用的模式。
contentVariable	Variable	绑定到图形项的 <code>content</code> 变量。绑定之前不必设置 <code>contentVariable</code> 的类型。Illustrator 会自动将类型设置为 <code>GRAPH</code> 。
controlBounds	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
editable	boolean	只读。如果为 <code>true</code> ，则此图形项是可编辑的。
geometricBounds	array of 4 numbers	只读。除笔划宽度以外的对象边界。
height	number (double)	图形项目的高度。
hidden	boolean	如果为 <code>true</code> ，则此图形项被隐藏。
isIsolated	boolean	如果为 <code>true</code> ，则隔离此对象。
layer	Layer	只读。该图形项所属的层。
left	number	图形项左侧与页面左侧的偏移量（以磅为单位）。
locked	boolean	如果为 <code>true</code> ，则此图形项被锁定。
name	string	该图项目的名称。
note	string	分配给该项目的注释。
opacity	number (double)	对象的不透明度；值介于 0.0 到 100.0 之间。
parent	Layer or GroupItem	只读。此对象的父级。
position	array of 2 numbers	<code>graphItem</code> 对象左上角的位置（以磅为单位），格式为 <code>[x, y]</code> 。不包括行程重量。
selected	boolean	如果为 <code>true</code> ，则选择此对象。
sliced	boolean	如果为 <code>true</code> ，则切片图项。默认值: <code>false</code>

属性	值类型	注释
tags	Tags	只读。此图形项中包含的标签。
top	number (double)	图表项目顶部与页面底部的偏移量（以磅为单位）。
typename	string	只读。图项目的类型。
uRL	string	分配给该图形项目的Adobe URL标记的值。
visibilityVariable	Variable	绑定到图形项的可见性变量。 绑定之前不必设置 visibleVariable 的类型。 Illustrator 会自动将类型设置为 VISIBILITY 。
visibleBounds	array of 4 numbers	只读。图形项目的可见范围，包括笔划宽度。
width	number (double)	图形项目的宽度。范围：0.0至16348.0
wrapInside	boolean	如果为 true ，则应将文本框架对象包装在该对象内。
wrapOffset	number (double)	在此对象上环绕文字时要使用的偏移量。
wrapped	boolean	如果为 true ，则将文本框架对象包装在该对象周围。（文本框必须在对象上方。）
zOrderPosition	number (long)	只读。该图稿的位置在包含该图稿的组或图层（父级）的堆叠顺序中。

GraphItem methods

方法	参数类型	返回值	作用
duplicate ([relativeObject] [, insertionLocation])	object ElementPlacement	GraphItem	创建所选对象的副本。
move (relativeObject, insertionLocation)	object ElementPlacement	GraphItem	移动对象。
remove ()		Nothing	删除该对象。

方法	参数类型	返回值	作用
resize (scaleX, scaleY [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, scaleAbout])	number (double) number (double) boolean boolean boolean number (double) Transformation	Nothing	缩放图稿，其中scaleX是水平缩放因子，而scaleY是垂直缩放因子。100.0 = 100%。
rotate (angle [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, rotateAbout])	number (double) boolean boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转图稿。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象。
transform (transformationMatrix [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, transformAbout])	Matrix boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换图稿。
translate ([deltaX] [, deltaY] [, transformObjects] [, transformFillPatterns] [, transformFillGradients] [, transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新放置图稿，其中delX是水平偏移，而delY是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列图稿的位置。

GraphItems

集合GraphItems对象，使您可以访问Illustrator文档中的所有图形图稿项目。

GraphItems properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

GraphItems methods

方法	参数类型	返回值	作用
getByName (name)	string	GraphItems	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	GraphItems	从集合中获取一个元素。
removeAll ()		Nothing	删除集合中的所有元素。

旋转图项目

```
//将当前文档中的每个图形项旋转90度。
//验证带有图形项的文档已打开
```

```
var ok = false;
if (documents.length > 0) {
  var docRef = activeDocument
  var iCount = docRef.graphItems.length
  if( iCount > 0) {
    ok = true;
    for (var i=0; i<iCount; i++) {
      var graphRef = docRef.graphItems[i];
      graphRef.selected = true;
      graphRef.rotate(90); //顺时针旋转90度
    }
    redraw();
  }
}
```

GrayColor

灰度颜色规范，用于需要颜色对象的地方。

GrayColor properties

属性	值类型	注释
gray	number (double)	灰色的色调。范围：0.0到100.0，其中0.0是黑色，而100.0是白色。
typename	string	只读。被引用对象的类名。

将颜色更改为灰色

```
//将活动文档中第一个单词的颜色
//设置为灰色阴影

if ( app.documents.length > 0
    && app.activeDocument.textFrames.length > 0 ) {
    var text = app.activeDocument.textFrames[0].textRange;
    var firstWord = text.words[0];

    //创建新颜色
    var textColor = new GrayColor();
    textColor.gray = 45;
    firstWord.filled = true;
    firstWord.fillColor = textColor;
}
```

GroupItem

一组图稿。组项目可以包含一个图层可以包含的所有相同页面项目，包括其他嵌套组。当脚本要求文档中包含的路径时，包含在文档组或复合路径中的路径将作为单独路径返回。但是，当脚本在包含组或复合路径的层中询问路径时，不会返回包含在组或复合路径中的路径。

GroupItem properties

属性	值类型	注释
<code>artworkKnockout</code>	KnockoutState	此对象用于创建淘汰对象吗？如果是，则使用哪种淘汰对象。
<code>blendingMode</code>	BlendModes	合成对象时使用的混合模式。
<code>clipped</code>	boolean	如果为 <code>true</code> ，则将组剪切到剪切蒙版。
<code>compoundPathItems</code>	CompoundPathItems	只读。此组中包含的复合路径项。
<code>controlBounds</code>	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
<code>editable</code>	boolean	只读。如果为 <code>true</code> ，则此项目是可编辑的。
<code>geometricBounds</code>	array of 4 numbers	只读。除笔划宽度以外的对象边界。
<code>graphItems</code>	GraphItems	只读。该组中包含的图形项。
<code>groupItems</code>	GroupItems	只读。该组中包含的组项目。
<code>height</code>	number (double)	组项目的高度。
<code>hidden</code>	boolean	如果为 <code>true</code> ，则此组项目是隐藏的。
<code>isIsolated</code>	boolean	如果为 <code>true</code> ，则隔离此对象。
<code>layer</code>	Layer	只读。该组项目所属的层。
<code>left</code>	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。
<code>legacyTextItems</code>	LegacyTextItems	只读。组中的旧文本项。
<code>locked</code>	boolean	如果为 <code>true</code> ，则此组项目被锁定。
<code>meshItems</code>	MeshItems	只读。此组中包含的网格物体。
<code>name</code>	string	该组项目的名称。
<code>nonNativeItems</code>	NonNativeItems	只读。该组中的非本地图稿作品。

属性	值类型	注释
<code>note</code>	string	分配给该项目的注释。
<code>opacity</code>	number (double)	对象的不透明度。范围：0.0至100.0
<code>pageItems</code>	PageItems	只读。该组中包含的页面项目（所有图稿项目类）。
<code>parent</code>	Layer or GroupItem	只读。此对象的父级。
<code>pathItems</code>	PathItems	只读。该组中包含的路径项。
<code>placedItems</code>	PlacedItems	只读。该组中包含的放置项目。
<code>pluginItems</code>	PluginItems	只读。该组中包含的插件项。
<code>position</code>	array of 2 numbers	<code>groupItem</code> 对象左上角的位置（以磅为单位），格式为[x, y]。不包括行程重量。
<code>rasterItems</code>	RasterItems	只读。此组中包含的栅格项目。
<code>selected</code>	boolean	如果为true，则选择此组项目。
<code>sliced</code>	boolean	如果为true，则将项目切成薄片。默认值：false
<code>symbolItems</code>	SymbolItems	只读。符号项目在该组中对象。
<code>tags</code>	Tags	只读。该组中包含的标签。
<code>textFrames</code>	TextFrameItems	只读。该组中包含的文字图稿项目。
<code>top</code>	number (double)	项目顶部的位置（以磅为单位，从页面底部开始测量）。
<code>typename</code>	string	只读。被引用对象的类名。
<code>uRL</code>	string	分配给该组项目的Adobe URL标记的值。
<code>visibilityVariable</code>	Variable	绑定到项目的可见性变量。
<code>visibleBounds</code>	array of 4 numbers	只读。组项目的可见范围，包括笔划宽度。
<code>width</code>	number (double)	群组项目的宽度。
<code>wrapInside</code>	boolean	如果为true，则应将文本框架对象包装在该对象内。
<code>wrapOffset</code>	number (double)	在此对象上环绕文字时要使用的偏移量。
<code>wrapped</code>	boolean	如果为true，则将文本框架对象包装在该对象周围（文本框架必须在该对象上方）。
<code>zOrderPosition</code>	number (long)	只读。该组对象在包含该组对象的组或层（父级）的堆叠顺序中的位置。

GroupItem methods

方法	参数类型	返回值	作用
duplicate ([relativeObject] [,insertionLocation])	object ElementPlacement	GroupItem	创建所选对象的副本。
move (relativeObject, insertionLocation)	object ElementPlacement	GroupItem	移动对象。
remove ()		Nothing	删除该对象。
resize (scaleX, scaleY [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,scaleAbout])	number (double) number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	缩放图稿，其中scaleX是水平缩放因子，而scaleY是垂直缩放因子。100.0 = 100%。
rotate (angle [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,rotateAbout])	number (double) boolean boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转图稿。如果角度值为正，则逆时针旋转对象；如果角度为负，则顺时针旋转对象。
transform (transformationMatrix [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,transformAbout])	Matrix boolean boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换图稿。
translate ([deltaX] [,deltaY] [,transformObjects] [,transformFillPatterns] [,transformFillGradients] [,transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新放置图稿，其中delX是水平偏移，而delY是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列图稿的位置。

修改组中的所有对象

修改组中包含的所有对象很容易。本示例演示如何通过创建包含多个对象的组来简化对多个对象的操作。

//创建一个新的组项目，向该组添加一个三角形的新路径项目，然后向该组中添加一个新的文本项目，并将文本的填充颜色设置为红色

```
if ( app.documents.length > 0 ) {
    var triangleGroup = app.activeDocument.groupItems.add();

    //创建一个三角形并添加文本，新的图稿作品将在组内创建
    var trianglePath = triangleGroup.pathItems.add();
    trianglePath.setEntirePath( Array( Array(100, 100), Array(300, 100),
        Array(200, Math.tan(1.0471975) * 100 + 100) ) );
    trianglePath.closed = true;
    trianglePath.stroked = true;
    trianglePath.filled = false;
    trianglePath.strokeWidth = 3;

    var captionText = triangleGroup.textFrames.add();
    captionText.position = Array(100, 150);
    captionText.textRange.size = 48;
    captionText.contents = "A triangle";

    var fillColor = new RGBColor;
    fillColor.red = 255;
    fillColor.green = 0;
    fillColor.blue = 0;
    captionText.characters.fillColor = fillColor;
}
```

GroupItems

文档中分组图稿的集合。

GroupItems properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

GroupItems methods

方法	参数类型	返回值	作用
add ()		GroupItem	创建一个新对象。
createFromFile (imageFile)	File	GroupItem	将外部矢量图稿文件作为组项目放置在文档中。
getByName (name)	string	GroupItem	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	GroupItem	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

将PDF导入为组项目

以下脚本显示了如何使用**createFromFile**函数导入PDF文档。在运行此脚本之前，您必须创建一页PDF文件并将其放在该位置

/temp/testfile1.pdf.

//从dest指定的文件中将新的组项目嵌入到当前文档中dest应该包含完整路径和文件名

```
function embedPDF(dest) {
    var embedDoc = new File(dest);
    if ( app.documents.length > 0 && embedDoc.exists ) {
        var doc = app.activeDocument;
        var placed = doc.groupItems.createFromFile( embedDoc );
    }
}
```

IllustratorSaveOptions

用于将文档另存为Illustrator文件的选项，与 [saveAs](#) 方法一起使用。所有属性都是可选的。

IllustratorSaveOptions properties

属性	值类型	注释
artboardRange	string	如果 saveMultipleArtboards 为true（仅对Illustrator 13或更早版本有效），则考虑对文档进行多资产提取，以指定画板范围。空字符串将提取所有画板。默认值：空字符串
compatibility	Compatibility	指定要创建的Illustrator文件格式的版本。默认值： <code>Compatibility.ILLUSTRATOR19</code>
compressed	boolean	（Illustrator 10或更高版本。）如果为true，则压缩保存的文件。默认值： <code>true</code>
embedICCPProfile	boolean	（Illustrator 9或更高版本。）如果为true，则文档的ICC配置文件将嵌入保存的文件中。默认值： <code>false</code>
embedLinkedFiles	boolean	（Illustrator 7或更高版本。）如果为true，则将链接的图像文件嵌入到已保存的文件中。默认： <code>false</code>
flattenOutput	OutputFlattening	（Illustrator 9之前的版本。）对于较旧的文件格式版本，应如何使透明度变平。默认值： <code>OutputFlattening.PRESERVEAPPEARANCE</code>
fontSubsetThreshold	number (double)	（Illustrator 9或更高版本。）在文档中使用少于此百分比字符时，请包括字体的子集。范围： <code>0.0</code> 到 <code>100.0</code> 。默认值： <code>100.0</code>
pdfCompatible	boolean	（Illustrator 10或更高版本。）如果为true，则将文件另存为PDF兼容文件。默认值： <code>true</code>
saveMultipleArtboards	boolean	如果为true，则保存所有画板或画板范围。对Illustrator 13或更早版本有效。
typename	string	只读。被引用对象的类名。

使用选项保存

//使用指定的选项将当前文档另存为dest作为AI文件,
// dest指定新文件的完整路径和文件名

```
function exportFileToAI (dest) {  
    if ( app.documents.length > 0 ) {  
        var saveOptions = new IllustratorSaveOptions();  
        var ai8Doc = new File(dest);  
        saveOptions.compatibility = Compatibility.ILLUSTRATOR8;  
        saveOptions.flattenOutput = OutputFlattening.PRESERVEAPPEARANCE;  
        app.activeDocument.saveAs( ai8Doc, saveOptions );  
    }  
}
```

ImageCaptureOptions

图像捕获选项，与 [imageCapture](#) 方法一起使用。所有属性都是可选的

ImageCaptureOptions properties

属性	值类型	注释
antiAliasing	boolean	如果为true，则图像结果是抗锯齿的。默认值: false
matte	boolean	如果为true，则画板将带有颜色。默认值: false
matteColor	RGBColor	画板遮罩使用的颜色。默认值: 白色
resolution	number (double)	捕获的图像文件的分辨率，以每英寸点数 (PPI) 为单位，范围为 [72.0 ... 2400.0]。默认值: 150
transparency	boolean	如果为true，则图像结果是透明的。默认值: false
typename	string	只读。被引用对象的类名。

Ink

将文档墨水名称与墨水信息相关联。

Ink properties

属性	值类型	注释
<code>inkInfo</code>	InkInfo	墨水信息
<code>name</code>	string	墨水名称
<code>typename</code>	string	只读。对象的类名

InkInfo

用于打印文档的墨水信息。

InkInfo properties

属性	值类型	注释
angle	number (double)	墨水的屏幕角度（以度为单位）。范围：-360至360
customColor	Color	自定义墨水的颜色。
density	number (double)	中性密度。最小值：0.0
dotShape	string	点形状名称。
frequency	number (double)	墨水的频率。范围：0.0至1000.0
kind	InkType	墨水类型。
printingStatus	InkPrintStatus	墨水打印状态。
trapping	TrappingType	陷阱类型。
trappingOrder	number (long)	陷阱的顺序。范围：CMYK为1到4
typename	string	只读。对象的类名称。

获取墨水信息

```
//在文本框中显示当前文档墨水

var docRef = documents.add();
var textRef = docRef.textFrames.add();

//在此文档中组装一串墨水
var sInks = "";
var iLength = activeDocument.inkList.length;

for(var i=0; i<iLength; i++) {
    sInks += docRef.inkList[i].name;
    sInks += "\r\t";
    sInks += "Frequency = " + docRef.inkList[i].inkInfo.frequency;
    sInks += "\r\t";
    sInks += "Density = " + docRef.inkList[i].inkInfo.density;
    sInks += "\r";
}
textRef.contents = sInks;
textRef.top = 600;
textRef.left = 200;
redraw();
```


InsertionPoint

字符之间的位置，用于插入新的文本对象。插入点包含在InsertionPoints集合中。

InsertionPoint properties

属性	值类型	注释
characters	Characters	只读。此文本范围内的所有字符。
lines	Lines	只读。此文本范围内的所有行。
paragraphs	Paragraphs	只读。此文本中的所有段落。
parent	TextRange	只读。对象的容器。
story	Story	只读。文字范围所属的故事。
textRanges	TextRanges	只读。此文本范围内的所有文本。
typename	string	只读。对象的类名。
words	Words	只读。该文本范围内包含的所有单词。

InsertionPoints插入点

InsertionPoint对象的集合。

InsertionPoints properties

属性	值类型	注释
length	number	只读。集合中元素的数量。
parent	object	只读。对象的容器。
typename	string	只读。对象的类名称。

InsertionPoints methods

方法	参数类型	返回值	作用
index (itemKey)	string, number	InsertionPoint	从集合中获取一个元素。

使用插入点添加空格

//创建一个新文档，添加文本，然后使用插入点在每个字符之间插入一个空格

```
var docRef = documents.add();
var textRef = docRef.textFrames.add();
textRef.contents = "Wouldn't you rather be scripting?";
textRef.top = 400;
textRef.left = 100;
textRef.textRange.characterAttributes.size = 20;
redraw();
```

//使用插入点在每个字符之间添加一个空格。

```
var ip;
for(var i=0; i<(textRef.insertionPoints.length); i+=2) {
  ip = textRef.insertionPoints[i];
  ip.characters.add(" ");
}
```

LabColor

CIE Lab颜色空间中的颜色规范，用于需要颜色对象的地方。

LabColor properties

属性	值类型	注释
a	number (double)	a (红绿色) 颜色值。范围-128.0-128.0。默认值: 0.0
b	number (double)	b (黄蓝色) 颜色值。范围-128.0-128.0。默认值: 0.0
l	number (double)	l (亮度) 颜色值。范围-128.0-128.0。默认值: 0.0

Layer

Illustrator文档中的图层。层可能包含嵌套层，这些嵌套层在用户界面中称为子层。

图层对象包含特定图层中的所有页面项目作为元素。您的脚本可以将页面项目作为Layer对象的元素或Document对象的元素进行访问。当将页面项作为图层的元素访问时，只能访问该图层中的对象。要访问整个文档中的页面项目，请确保引用文档中包含的页面项目。

Layer properties

属性	值类型	注释
<code>artworkKnockout</code>	KnockoutState	此对象用于创建淘汰对象吗？如果是，则使用哪种淘汰对象。您不能将此值设置为 <code>KnockoutState.Unknown</code> 。
<code>blendingMode</code>	BlendModes	合成对象时使用的模式。
<code>color</code>	RGBColor	图层的选择标记颜色。
<code>compoundPathItems</code>	CompoundPathItems	只读。此层中包含的复合路径项。
<code>dimPlacedImages</code>	boolean	如果为 <code>true</code> ，则在此层中应将放置的图像渲染为暗淡的。
<code>graphItems</code>	GraphItems	只读。此层中包含的图形项。
<code>groupItems</code>	GroupItems	只读。该层中包含的组项目。
<code>hasSelectedArtwork</code>	boolean	如果为 <code>true</code> ，则已选择此层中的一个对象；否则为 <code>0</code> 。设置为 <code>false</code> 可取消选择图层中的所有对象。
<code>isIsolated</code>	boolean	如果为 <code>true</code> ，则隔离此对象。
<code>layers</code>	Layers	只读。该层中包含的层。
<code>legacyTextItems</code>	LegacyTextItems	只读。此层中的旧文本项。
<code>locked</code>	boolean	如果为 <code>true</code> ，则此层是可编辑的。设置为 <code>false</code> 以锁定图层。
<code>meshItems</code>	MeshItems	只读。此层中包含的网格项。
<code>name</code>	string	该层的名称。
<code>nonNativeItems</code>	NonNativeItems	这一层中的非本地图稿作品。
<code>opacity</code>	number (double)	图层的透明度。范围：0.0至100.0
<code>pageItems</code>	PageItems	只读。此层中包含的页面项目（所有图稿项目类）。
<code>parent</code>	Document Or Layer	只读。包含该层的文档或层。

属性	值类型	注释
<code>pathItems</code>	PathItems	只读。此层中包含的路径项。
<code>placedItems</code>	PlacedItems	只读。此层中包含的已放置项目。
<code>pluginItems</code>	PluginItems	只读。此层中包含的插件项。
<code>preview</code>	boolean	如果为true，则应使用预览模式显示此层。
<code>printable</code>	boolean	如果为true，则在打印文档时应打印此层。
<code>rasterItems</code>	RasterItems	只读。此图层中包含的栅格项。
<code>sliced</code>	boolean	如果为true，则切片项。默认值: false
<code>symbolItems</code>	SymbolItems	只读。图层中包含的符号项目。
<code>textFrames</code>	TextFrameItems	只读。此图层中包含的文字图稿项目。
<code>typename</code>	string	只读。被引用对象的类名。
<code>visible</code>	boolean	如果为true，则此层可见。
<code>zOrderPosition</code>	number (long)	只读。该层在文档中层的堆叠顺序中的位置。

Layer methods

方法	参数类型	返回值	作用
<code>move</code> (relativeObject, insertionLocation)	object ElementPlacement	Layer	移动对象。
<code>remove</code> ()		Nothing	删除该对象。
<code>zOrder</code> (ZOrderCmd)	ZOrderMethod	Nothing	按照该对象的包含层或文档（父级）的堆叠顺序排列层的位置

在前面放一层

//移动最底层成为最顶层

```
if (documents.length > 0) {  
    countOfLayers = activeDocument.layers.length;  
    if (countOfLayers > 1) {  
        bottomLayer = activeDocument.layers[countOfLayers-1];  
        bottomLayer.zOrder(ZOrderMethod.BRINGTOFRONT);  
    }  
    else {  
        alert("The active document only has only 1 layer")  
    }  
}
```

Layers

文档中图层的集合。

Layers properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

Layers methods

方法	参数类型	返回值	作用
add ()		Layer	在文档中创建一个新层。
getByName (name)	string	Layer	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	Layer	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

查找和删除图层

//在所有打开的文档中删除所有名称以“Temp”开头的图层
//在所有打开的文档中循环

```
var layersDeleted = 0;
for ( i = 0; i < app.documents.length; i++ ) {
    var targetDocument = app.documents[i];
    var layerCount = targetDocument.layers.length;
    // Loop through layers from the back, to preserve index
    // of remaining layers when we remove one
    for (var ii = layerCount - 1; ii >= 0; ii-- ) {
        targetLayer = targetDocument.layers[ii];
        var layerName = new String( targetLayer.name );
        if ( layerName.indexOf("Temp") == 0 ) {
            targetDocument.layers[ii].remove();
            layersDeleted++;
        }
    }
}
}
```

LegacyTextItem

在Illustrator CS（版本10）或更早版本中创建的文本对象，在转换之前不可编辑。要转换旧文本，请参见 [convertToNative](#)。

您可以查看，移动和打印旧文本，但不能对其进行编辑。选中时，旧文本的边界框内有一个“x”。

LegacyTextItem properties

属性	值类型	注释
artworkKnockout	KnockoutState	此对象用于创建淘汰对象吗？如果是，则使用哪种淘汰对象。
blendingMode	BlendModes	合成对象时使用的混合模式。
controlBounds	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
converted	boolean	只读。如果为true，则旧文本项目已更新为本地文本框架项目。
editable	boolean	只读。如果为true，则此项目是可编辑的。
geometricBounds	array of 4 numbers	只读。除笔划宽度以外的对象边界。
height	number (double)	组项目的高度。
hidden	boolean	如果为true，则此项是隐藏的。
isIsolated	boolean	如果为true，则隔离此对象。
layer	Layer	只读。此项所属的图层。
left	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。
locked	boolean	如果为true，则该项被锁定。
name	string	此项的名称。
note	string	分配给该项目的注释。
opacity	number (double)	对象的不透明度。范围：0.0至100.0
parent	Layer Or GroupItem	只读。此对象的父级。
position	array of 2 numbers	legacyTextItem 对象的左上角的位置（以磅为单位），格式为[x, y]。
selected	boolean	如果为true，则选择此项目。
sliced	boolean	如果为true，则将项目切成薄片。默认值：false
tags	Tags	只读。此项目中包含的标签。

属性	值类型	注释
top	number (double)	项目顶部的位置（以磅为单位，从页面底部开始测量）。
typename	string	只读。被引用对象的类名。
uRL	string	分配给此项的Adobe URL标记的值。
visibilityVariable	Variable	绑定到项目的可见性变量。
visibleBounds	array of 4 numbers	只读。项目的可见边界，包括笔触宽度。
width	number (double)	项目的宽度。
wrapInside	boolean	如果为true，则应将文本框架对象包装在该对象内。
wrapOffset	number (double)	在此对象上环绕文字时要使用的偏移量。
wrapped	boolean	如果为true，则将文本框架对象包装在该对象周围（文本框架必须在该对象上方）。
zOrderPosition	number (long)	只读。该项目在包含该项目的组或层（父级）的堆叠顺序中的位置。

LegacyTextItem methods

方法	参数类型	返回值	作用
convertToNative ()		GroupItem	将旧式文本项转换为文本框架并删除原始旧式文本。
duplicate ([relativeObject] [, insertionLocation])	object ElementPlacement	LegacyTextItem	创建所选对象的副本。
move (relativeObject, insertionLocation)	object ElementPlacement	LegacyTextItem	移动对象。
remove ()		Nothing	删除该对象。

方法	参数类型	返回值	作用
resize (scaleX, scaleY [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,scaleAbout])	number (double) number (double) boolean boolean boolean number (double) Transformation	Nothing	缩放图稿，其中 scaleX 是水平缩放因子，而 scaleY 是垂直缩放因子。100.0 = 100%。
rotate (angle [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,rotateAbout])	number (double) boolean boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转图稿。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象。
transform (transformationMatrix [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,transformAbout])	Matrix boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换图稿。
translate ([deltaX] [,deltaY] [,transformObjects] [,transformFillPatterns] [,transformFillGradients] [,transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新放置图稿，其中 delX 是水平偏移，而 delY 是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列图稿的位置。

LegacyTextItems

LegacyTextItem对象的集合。

LegacyTextItems properties

属性	值类型	注释
length	number	只读。集合中元素的数量。
parent	object	只读。对象的容器。
typename	string	只读。对象的类名称。

LegacyTextItems methods

方法	参数类型	返回值	作用
convertToNative ()		boolean	从所有旧有文本项创建文本框架；原始旧文本项将被删除。成功返回 true 。
getByName (name)	string	LegacyTextItem	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	LegacyTextItem	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

Lines

表示文本行在文本框架中的TextRange对象的集合。元素未命名；您必须按索引访问它们。

Lines properties

属性	值类型	注释
length	number	只读。集合中元素的数量。
parent	object	只读。对象的容器。
typename	string	只读。对象的类名称。

Lines methods

方法	参数类型	返回值	作用
index (itemKey)	number	TextRange	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

Matrix

转换矩阵规范，用于转换对象的几何形状。使用它可以从Illustrator文档或文档中的页面项目指定和检索矩阵信息。

Matrix与transform方法结合使用，并作为许多对象的属性。矩阵指定如何变换对象的几何形状。您可以使用Application对象方法getTranslationMatrix，getScaleMatrix或getRotationMatrix生成原始矩阵。

矩阵是包含矩阵值的记录，而不是对矩阵对象的引用。矩阵命令对矩阵记录的值进行操作。如果命令修改了矩阵，则返回修改后的矩阵记录作为命令的结果。传递给命令的原始矩阵记录不会被修改。

Matrix properties

属性	值类型	注释
mValueA	number (double)	矩阵属性 a.
mValueB	number (double)	矩阵属性 b.
mValueC	number (double)	矩阵属性 c.
mValueD	number (double)	矩阵属性 d.
mValueTX	number (double)	矩阵属性 tx.
mValueTY	number (double)	矩阵属性 ty.
typename	string	只读。被引用对象的类名。

组合矩阵以应用多个变换

要将多个变换应用于对象，使用矩阵套件要比一次应用一个变换更为有效。以下脚本演示了如何组合多个矩阵。

```
//使用平移和旋转矩阵对文档中的所有图稿进行变
形，//将图稿向右移动半英寸，在页面上向上移动1.5英
寸

if (app.docMatrices.length > 0) {
    var moveMatrix = app.doc.getTranslationMatrix( 0.5, 1.5 );
    //向平移添加旋转，逆时针旋转10度
    var totalMatrix = concatenateRotationMatrix( moveMatrix,
    10 ); //将转换应用于文档中的所有图稿
    var doc = app.activeDocument;
    for ( i = 0; i < doc.pageItems.length; i++ ) {
        doc.pageItems[i].transform( totalMatrix );
    }
}
```

MeshItem

渐变网格图稿。您无法通过脚本创建网格物体。但是，您可以使用重复方法复制现有的网格物体，然后使用其中一种移动方法将副本放置在正确的位置。

MeshItem properties

属性	值类型	注释
artworkKnockout	KnockoutState	此对象用于创建蒙版对象吗？如果是，则使用哪种蒙版对象。
blendingMode	BlendModes	合成对象时使用的混合模式。
controlBounds	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
editable	boolean	只读。如果为true，则此项目是可编辑的。
geometricBounds	array of 4 numbers	只读。除笔划宽度以外的对象边界。
height	number (double)	组项目的高度。
hidden	boolean	如果为true，则此项是隐藏的。
isIsolated	boolean	如果为true，则隔离此对象。
layer	Layer	只读。此项所属的图层。
left	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。
locked	boolean	如果为true，则该项被锁定。
name	string	此项的名称。
note	string	分配给该项目的注释。
opacity	number (double)	对象的不透明度。范围：0.0至100.0
parent	Layer or GroupItem	只读。此对象的父级。
position	array of 2 numbers	meshItem对象左上角的位置（以磅为单位），格式为[x, y]。不包括行程重量。
selected	boolean	如果为true，则选择此项目。
sliced	boolean	如果为true，则将项目切成薄片。默认值: false
tags	Tags	只读。此项目中包含的标签。
top	number (double)	项目顶部的位置（以磅为单位，从页面底部开始测量）。

属性	值类型	注释
<code>typename</code>	string	只读。被引用对象的类名。
<code>uRL</code>	string	分配给此项的Adobe URL标记的值。
<code>visibilityVariable</code>	Variable	绑定到项目的可见性变量。
<code>visibleBounds</code>	array of 4 numbers	只读。项目的可见边界，包括笔触宽度。
<code>width</code>	number (double)	项目的宽度。
<code>wrapInside</code>	boolean	如果为true，则应将文本框架对象包装在该对象内。
<code>wrapOffset</code>	number (double)	在此对象上环绕文字时要使用的偏移量。
<code>wrapped</code>	boolean	如果为true，则将文本框架对象包装在该对象周围（文本框架必须在该对象上方）。
<code>zOrderPosition</code>	number (long)	只读。该项目在包含该项目的组或层（父级）的堆叠顺序中的位置。

MeshItem methods

方法	参数类型	返回值	作用
<code>duplicate</code> ([relativeObject] [, insertionLocation])	object ElementPlacement	MeshItem	创建所选对象的副本。
<code>move</code> (relativeObject, insertionLocation)	object ElementPlacement	MeshItem	移动对象。 不是位置移动
<code>remove</code> ()		Nothing	删除该对象。
<code>resize</code> (scaleX, scaleY [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, scaleAbout])	number (double) number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	缩放艺术品，其中scaleX是水平缩放因子，而scaleY是垂直缩放因子。100.0 = 100%。
<code>rotate</code> (angle [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, rotateAbout])	number (double) boolean boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转艺术品。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象。

方法	参数类型	返回值	作用
transform (transformationMatrix [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,transformAbout])	Matrix boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换艺术品。
translate ([deltaX] [,deltaY] [,transformObjects] [,transformFillPatterns] [,transformFillGradients] [,transformStrokePatterns])	number (double) number (double) boolean boolean boolean	Nothing	相对于当前位置重新放置艺术品，其中delX是水平偏移，而delY是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列艺术品的位置。

查找并锁定网格物体

//锁定当前文档中的所有网格项

```

if ( app.documents.length > 0 ) {
  doc = app.activeDocument;
  for ( i = 0; i < doc.meshItems.length; i++ ) {
    doc.meshItems[i].locked = true;
  }
}

```


MeshItems

MeshItem对象的集合。

MeshItems properties

属性	值类型	注释
length	number	只读。集合中的对象数
parent	object	只读。该对象的父对象
typename	string	只读。被引用对象的类名。

MeshItems methods

方法	参数类型	返回值	作用
getByName (name)	string	MeshItem	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	MeshItem	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

将网格项目复制到另一个文档

要运行此脚本，请打开两个文档。一个文档应至少包含一个网格项目，另一文档可以为空。在运行脚本之前，将空文档放在最前面。

```
//将所有网格物体项目从一个文档复制到新文档
```

```
if ( app.documents.length > 0 ) {
    var srcDoc = documents[0];
    var locationOffset = 0;
    var targetDoc = documents.add();

    for ( i = 0; i < srcDoc.meshItems.length; i++) {
        srcItem = srcDoc.meshItems[i];
        var dupItem = srcDoc.meshItems[i].duplicate( targetDoc,
            ElementPlacement.PLACEATEND );

        // offset the copied items' position on the y axis
        dupItem.position = Array( 100, 50 + locationOffset );
        locationOffset += 50;
    }
}
```

NoColor

代表“无”颜色。将NoColor对象分配给艺术作品的填充或描边颜色等同于将填充或描边属性设置为false。

NoColor properties

属性	值类型	注释
typename	string	只读。对象的类名

使用NoColor删除填充颜色

```
//创建2个具有不同填充颜色的重叠对象。
```

```
//为顶部对象分配“ NoColor”填充色
```

```
//使底部对象可见。
```

```
//创建2个重叠的对象，一个为蓝色，一个为红色；  
var docRef = documents.add();  
var itemRef1 = docRef.pathItems.rectangle(500, 200, 200, 100);  
var itemRef2 = docRef.pathItems.rectangle(550, 150, 200, 200);  
var rgbColor = new RGBColor();  
rgbColor.red = 255;  
itemRef2.fillColor = rgbColor;  
rgbColor.blue = 255;  
rgbColor.red = 0;  
itemRef1.fillColor = rgbColor;  
redraw();
```

```
//创建一个nocolor并将其分配给顶部对象  
var noColor = new NoColor();  
itemRef2.fillColor = noColor;  
redraw();
```

NonNativeItem

非本地艺术品。

NonNativeItem properties

这些类继承了页面项目类的所有属性。

属性	值类型	注释
artworkKnockout	KnockoutState	此对象用于创建蒙版对象吗？如果是，则使用哪种蒙版类型。
blendingMode	BlendModes	合成对象时使用的混合模式。
controlBounds	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
editable	boolean	只读。如果为true，则此项目是可编辑的。
geometricBounds	array of 4 numbers	只读。项目的可见边界，包括笔触宽度。
height	number (double)	组项目的高度。
hidden	boolean	如果为true，则此项是隐藏的。
isIsolated	boolean	如果为true，则隔离此对象。
layer	Layer	只读。此项所属的图层。
left	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。
locked	boolean	如果为true，则该项被锁定。
name	string	此项的名称。
note	string	该项目的注释。
opacity	number (double)	对象的不透明度。范围：0.0至100.0
parent	Document , Layer , or GroupItem	只读。此对象的父级。
position	array of 2 numbers	NonNativeItem对象左上角的位置（以磅为单位），格式为[x, y]。不包括行程重量。
selected	boolean	如果为true，则选择此项目。
sliced	boolean	如果为true，则将项目选中。默认值：false
tags	Tags	只读。此项目中包含的标签。
top	number (double)	项目顶部的位置（以磅为单位，从页面底部开始测量）。

属性	值类型	注释
typename	string	只读。被引用对象的类名。
uRL	string	分配给此项的Adobe URL标记的值。
visibilityVariable	Variable	绑定到项目的可见性变量。
visibleBounds	array of 4 numbers	只读。项目的可见边界，包括笔触宽度。
width	number (double)	项目的宽度。
wrapInside	boolean	如果为true，则应将非本机对象包装在该对象内。
wrapOffset	number (double)	在此对象上环绕文字时要使用的偏移量。
wrapped	boolean	如果为true，请在该对象周围包装非本机对象（非本机对象必须在该对象上方）。
zOrderPosition	number	只读。该项目在包含该项目的组或层（父级）的堆叠顺序中的位置。

NonNativeItem methods

方法	参数类型	返回值	作用
duplicate ([relativeObject] [, insertionLocation])	object ElementPlacement	NonNativeItem	创建所选对象的副本。
move (relativeObject, insertionLocation)	object ElementPlacement	NonNativeItem	移动对象。
remove ()		Nothing	删除该对象。
removeAll ()		Nothing	删除此集合中的所有元素。
resize (scaleX, scaleY [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, scaleAbout])	number (double) number (double) boolean boolean boolean number (double) Transformation	Nothing	缩放艺术品，其中scaleX是水平缩放因子，而scaleY是垂直缩放因子。100.0 = 100%。

方法	参数类型	返回值	作用
rotate (angle [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, rotateAbout])	number (double) boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转艺术品。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象。
transform (transformationMatrix [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, transformAbout])	Matrix boolean boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换艺术品。
translate ([deltaX] [, deltaY] [, transformObjects] [, transformFillPatterns] [, transformFillGradients] [, transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新放置艺术品，其中delX是水平偏移，而delY是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列艺术品的位置。

NonNativeItems

A collection of `NonNativeItem` objects.

NonNativeItems properties

属性	值类型	注释
<code>length</code>	<code>number</code>	只读。集合中的对象数。
<code>parent</code>	<code>object</code>	只读。此对象的父级。
<code>typename</code>	<code>string</code>	只读。被引用对象的类名。

NonNativeItems methods

方法	参数类型	返回值	作用
<code>getByName</code> (name)	<code>string</code>	NonNativeItem , SymbolItem	获取具有指定名称的集合中的第一个元素。

OpenOptions

用于打开文档的选项，与 [open](#) 方法一起使用。

OpenOptions properties

属性	值类型	注释
<code>convertCropAreaToArboard</code>	boolean	可选的。在Illustrator CS4或更高版本中打开旧文档时，将裁剪区域转换为画板。如果为假，则丢弃作物面积。默认值： <code>true</code> 。
<code>convertTilesToArboard</code>	boolean	可选的。在Illustrator CS4或更高版本中打开旧文档时，将打印图块转换为画板。默认值： <code>false</code> 。
<code>createArtboardWithArtworkBoundingBox</code>	boolean	可选的。在Illustrator CS4或更高版本中打开旧文档时，请使用图稿边框的尺寸创建一个画板。默认值： <code>false</code> 。
<code>openAs</code>	LibraryType	可选的。打开该文件作为此类型的Illustrator库。默认值： <code>LibraryType.IllustratorArtwork</code> 。
<code>preserveLegacyArtboard</code>	boolean	可选的。在Illustrator CS4或更高版本中打开旧版文档时，保留旧版画板。默认值： <code>true</code> 。
<code>updateLegacyGradientMesh</code>	boolean	如果为true，则在旧文档的渐变网格对象中保留专色（Illustrator CS4之前的版本）。默认值： <code>true</code>
<code>updateLegacyText</code>	boolean	可选的。如果为true，则更新所有旧文本项（来自Illustrator的早期版本）。默认值： <code>false</code>

打开时自动更新旧文本

//使用OpenOptions自动更新包含旧文本的文件（AI 10或更早版本）。

```
var fileRef = filePath;
if (fileRef != null) {
    var optRef = new OpenOptions();
    optRef.updateLegacyText = true;
    var docRef = open(fileRef, DocumentColorSpace.RGB, optRef);
}
```

OpenOptionsAutoCAD

用于打开AutoCAD图形的选项，与 [open](#) 方法一起使用。

OpenOptionsAutoCAD properties

属性	值类型	注释
centerArtwork	boolean	如果为true，则艺术品在画板上居中。 默认值: true
globalScaleOption	AutoCADGlobalScaleOption	如何在导入时缩放工程图。默认值: AutoCAD全局比例选项。
globalScalePercent	double	globalScaleOption 为 AutoCADGlobalScaleOption.ScaleByValue 时的值，以百分比表示。范围: 0.0到100.0。默认值为100.0
mergeLayers	boolean	如果为true，则合并图稿的图层。默认值: false
parent	object	只读。对象的容器。
scaleLineweights	boolean	如果为true，则线宽将按与图形其余部分相同的比例缩放。 默认值: false
selectedLayoutName	string	要导入的工程图中的布局名称。
typename	string	只读。对象的类名称。
unit	AutoCADUnit	要映射的单位。 默认值: AutoCADUnit.Millimeters
unitScaleRatio	double	映射单位时缩放的比例。默认值: 1.0

OpenOptionsFreeHand

用于打开FreeHand文件的选项。

OpenOptionsFreeHand properties

属性	值类型	注释
<code>convertTextToOutlines</code>	boolean	如果为true, 则所有文本都将转换为矢量路径; 保留字体的外观。默认值: false
<code>importSinglePage</code>	boolean	如果为true, 则仅导入pageToOpen属性中指定的页面。默认值: true
<code>pageToOpen</code>	long	打开多页文档时要导入的页码。仅在importSinglePage为true时有效。
<code>parent</code>	object	只读。此对象的父级。
<code>typename</code>	string	只读。被引用对象的类名。

OpenOptionsPhotoshop

用于打开Photoshop文档的选项，与 [open](#) 方法一起使用。

OpenOptionsPhotoshop properties

属性	值类型	注释
layerComp	string	转换文档时要使用的图层复合的名称。
preserveHiddenLayers	boolean	如果为true，则在转换文档时保留隐藏的图层。默认值：false。
preserveImageMaps	boolean	如果为true，则在转换文档时保留图像映射。默认值：true。
preserveLayers	boolean	如果为true，则在转换文档时保留图层。默认值：true。
preserveSlices	boolean	如果为true，则在转换文档时保留切片。默认值：true。
typename	string	只读。对象的类名称。

PageItem

任何艺术品。文档中的每个美术项目和组都是一个页面项目。您可以将页面项目称为文档，图层或组项目的元素。

PageItem类使您可以完全访问Illustrator文档中包含的每个艺术品。PageItem类是文档中所有图稿对象的超类。CompoundPathItem, GroupItem, MeshItem, PathItem, PlacedItem, PluginItem, RasterItem和TextFrame类每个都从PageItem类继承一组属性。

您不能直接创建PageItem，必须创建特定的PageItem子类之一，例如PathItem。

PageItem properties

属性	值类型	注释
artworkKnockout	KnockoutState	该对象用于创建蒙版吗？
blendingMode	BlendModes	合成此对象时使用的模式。如果对象的不透明度设置为小于100.0（100%），则认为该对象已合成。
controlBounds	rect	只读。对象的边界，包括笔触宽度和控件。
editable	boolean	只读。如果为true，则此页面项目是可编辑的。
geometricBounds	rect	只读。对象的边界（不包括笔触宽度）。
height	real	页面项目的高度，根据几何范围计算得出。范围：0.0至16348.0
hidden	boolean	如果为true，则此页面项是隐藏的。
isIsolated	boolean	如果为true，则隔离此对象。
layer	Layer	只读。此页面项所属的层。
left	number (double)	艺术作品的左侧位置。
locked	boolean	如果为true，则此页面项目被锁定。
name	string	此页面项目的名称。
note	string	该项目的注释。
opacity	real	此对象的不透明度，其中100.0是完全不透明的，而0.0是完全透明的。
parent	object	只读。此对象的父级。
pixelAligned	boolean	如果此项与像素网格对齐，则为True。
position	point	项目左上角的位置（以磅为单位），格式为{x, y}。不包括行程重量。

属性	值类型	注释
selected	boolean	如果为true，则选择此对象。
sliced	boolean	如果为true，则保留切片。
tags	Tags	与此页面项目关联的标签的集合。
top	number (double)	艺术作品的最高位置。
typename	string	只读。对象的类名称。
URL	string	分配给此页面项目的Adobe URL标记的值。
visibilityVariable	anything	此页面项目路径绑定到的可见性变量。
visibleBounds	rect	只读。对象的可见范围，包括插图中任何对象的笔划宽度。
width	real	页面项目的宽度，根据几何边界计算得出。范围：0.0至16348.0
wrapInside	boolean	如果为true，则应将文本框架对象包装在该对象内。
wrapOffset	number (double)	在此对象上环绕文字时要使用的偏移量。
wrapped	boolean	如果为true，则将文本框架对象包装在该对象周围（文本框架必须在该对象上方）。
zOrderPosition	number (long)	只读。艺术作品在其组或图层中的绘制顺序。

PageItem methods

方法	参数类型	返回值	作用
bringInPerspective (posX, posY, perspectiveGridPlane)	number number PerspectiveGridPlaneType	Nothing	将艺术品放置在透视网格中指定位置和网格平面上。
resize (scaleX, scaleY [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,scaleAbout])	number (double) number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	缩放艺术品。

方法	参数类型	返回值	作用
rotate (angle [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,rotateAbout])	number (double) boolean boolean boolean Transformation	Nothing	旋转艺术品。
transform (transformationMatrix [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidth] [,transformAbout])	matrix boolean boolean boolean boolean number (double) Transformation	Nothing	使用转换矩阵对艺术品进行转换。
translate ([deltaX] [,deltaY] [,transformObjects] [,transformFillPatterns] [,transformFillGradients] [,transformStrokePattern])	number (double) number (double) boolean boolean boolean boolean	Nothing	重新放置艺术品。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	相对于组或图层中的其他技术来排列技术。

PageItems

页面项目对象的集合。提供对以下类的Illustrator文档中的所有艺术品的完整访问：

[CompoundPathItem](#)

[GraphItem](#)

[GroupItem](#)

[LegacyTextItem](#)

[MeshItem](#)

[NonNativeItem](#)

[PathItem](#)

[PlacedItem](#)

[PluginItem](#)

[RasterItem](#)

[SymbolItem](#)

[TextFrameItem](#)

您可以通过文档，图层或组中的PageItems属性引用页面项目。当您访问这些集合之一中的单个项目时，引用就是特定类型之一的页面项目。例如，如果使用PageItems引用图形项目，则该对象的typename值为GraphItem。

PageItems properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

PageItems methods

方法	参数类型	返回值	作用
getByName (name)	string	PageItem	获取具有指定名称的集合中的一个元素。
index (itemKey)	string, number	PageItem	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

在页面项目中获取对外部文件的引用

运行此脚本之前，请打开包含一个或多个链接图像的文档。

//使用pageItems对象获取当前文档中的所有文件引用，然后将其显示在新文档中

```
if ( app.documents.length > 0 ) {  
  
    var fileReferences = new Array();  
  
    var sourceDoc = app.activeDocument;  
    var sourceName =sourceDoc.name;  
    for ( i = 0; i < sourceDoc.pageItems.length; i++ ) {  
        artItem = sourceDoc.pageItems[i];  
        switch ( artItem.typename ) {  
            case "PlacedItem":  
                fileReferences.push( artItem.file.fsName );  
                break;  
            case "RasterItem":  
                if ( ! artItem.embedded ) {  
                    fileReferences.push( artItem.file.fsName );  
                }  
                break;  
        }  
    }  
    // Write the file references to a new document  
    var reportDoc = documents.add();  
    var areaTextPath = reportDoc.pathItems.rectangle( reportDoc.height,0,  
        reportDoc.width, reportDoc.height );  
    var fileNameText = reportDoc.textFrames.areaText( areaTextPath );  
    fileNameText.textRange.size = 24;  
    var paragraphCount = 3;  
    var text = "File references in '\" + sourceName + "\':\r\r";  
    for ( i = 0; i < fileReferences.length; i++ ) {  
        text += ( fileReferences[i] + "\r" );  
        paragraphCount++;  
    }  
    fileNameText.contents = text;  
}
```

Paper

将纸张信息与纸张名称关联。纸张对象由打印机对象使用。

Paper properties

属性	值类型	注释
name	string	纸张名称。
paperInfo	PaperInfo	纸张信息。
typename	string	只读。对象的类名称。

PaperInfo

用于打印文档的纸张信息。

PaperInfo properties

属性	值类型	注释
customPaper	boolean	如果为true, 则为自定义纸张。
height	number (double)	纸张的高度 (以磅为单位)。
imageableArea	array of 4 numbers	可成像区域。
typename	string	只读。对象的类名称。
width	number (double)	The paper's width in points.

查找纸张信息

//在文本框中显示第二台打印机可用的纸张和纸张尺寸

```

var docRef = documents.add();
var itemRef = docRef.pathItems.rectangle(600, 300, 200, 100);
var textRef = docRef.textFrames.add();
textRef.top = 600;
textRef.left = 50;
//获取第二台打印机的纸张对象
var printerRef = printerList[1];
textRef.contents = printerRef.name;
textRef.contents += " paper list:\r";
var paragraphCount = 2;
//获取每篇论文的详细信息
var iCount = printerRef.printerInfo.paperSizes.length;
for( var i=0; i<iCount; i++ ) {
    var paperRef = printerRef.printerInfo.paperSizes[i];
    var paperInfoRef = paperRef.paperInfo;
    textRef.contents += paperRef.name;
    textRef.contents += "\t";
    textRef.contents += paperInfoRef.height;
    textRef.contents += " x ";
    textRef.contents += paperInfoRef.width;
    textRef.contents += "\r";
    paragraphCount++;
}
redraw();

```

ParagraphAttributes

指定文本框中包含的段落的属性和属性。

注意：段落属性没有默认值，并且在明确设置之前是未定义的。

ParagraphAttributes properties

属性	值类型	注释
autoLeadingAmount	number (double)	Auto leading amount expressed as a percentage.
bunriKinshi	boolean	如果为true，则启用BunriKinshi。
burasagariType	BurasagariTypeEnum	Burasagari类型。
desiredGlyphScaling	number (double)	所需的字形缩放比例，以默认字符宽度的百分比表示。范围：50.0至200.0。在100.0时，字符的宽度不变。
desiredLetterSpacing	number (double)	所需字母，间距表示为默认字距调整或跟踪的百分比范围：-100.0至500.0。为0时，字母之间不添加空格。在100.0时，在字母之间添加整个空格宽度。
desiredWordSpacing	number (double)	所需字距，表示为字体默认空间的百分比。范围：0.0至1000.0；在100.00。单词之间没有空格。
everyLineComposer	boolean	如果为true，则启用Every-line Composer。如果为false，则启用Single-line Composer。
firstLineIndent	number (double)	First line left indent in points.
hyphenateCapitalizedWords	boolean	如果为true，则为大写单词启用连字符。
hyphenation	boolean	如果为true，则为段落启用连字符。
hyphenationPreference	number (double)	连字符首选项标度用于更好的间距（0）或更少的连字符（1）。范围：0.0至1.0

属性	值类型	注释
<code>hyphenationZone</code>	number (double)	距段落右边缘的距离（以磅为单位），该距离标记了行中不允许断字的部分。 NOTE: 0 allows all hyphenation. Valid only when everyLineComposer is false.
<code>justification</code>	Justification	段落对方样式。
<code>kinsoku</code>	string	Kinsoku Shori名称。
<code>kinsokuOrder</code>	KinsokuOrderEnum	首选的Kinsoku订单。
<code>kurikaeshiMojishori</code>	boolean	如果为true，则启用KurikaeshiMojishori。
<code>leadingType</code>	AutoLeadingType	自动引导类型。
<code>leftIndent</code>	number (double)	页边距的左缩进点数。
<code>maximumConsecutiveHyphens</code>	number (long)	连续连字符行的最大数量。
<code>maximumGlyphScaling</code>	number (double)	最大字形缩放比例，以默认字符宽度的百分比表示。 范围：50.0至200.0；在100.0。 字符的宽度不变。 注意：仅对合理的段落有效。
<code>maximumLetterSpacing</code>	number (double)	最大字母间距，表示为默认字距调整或跟踪的百分比。范围：-100.0到500.0；默认值为255。为0。字母之间没有空格。在100.0时，在字母之间添加整个空格宽度。 注意：仅对合理的段落有效。
<code>maximumWordSpacing</code>	number (double)	最大字间距，以字体默认空间的百分比表示。 范围：0.0至1000.0；在100.00。 单词之间没有空格。 注意：仅对合理的段落有效。
<code>minimumAfterHyphen</code>	number (long)	连字符后的最小字符数。
<code>minimumBeforeHyphen</code>	number (long)	连字符前的最小字符数。

属性	值类型	注释
minimumGlyphScaling	number (double)	最小字形缩放，以默认字符宽度的百分比表示。范围：50.0至200.0。在100.0时，字符的宽度不变。 注意：仅对合理的段落有效。
minimumHyphenatedWordSize	number (long)	要连字的单词的最小字符数。
minimumLetterSpacing	number (double)	最小字母间距，表示为默认字距调整或跟踪的百分比。范围：-100.0至500.0；默认值为0.0。为0。字母之间没有空格。在100.0时，在字母之间添加整个空格宽度。 注意：仅对合理的段落有效。
minimumWordSpacing	number (double)	最小字距，表示为字体默认空间的百分比。 范围：0.0至1000.0；在100.00。 单词之间没有空格。 注意：仅对合理的段落有效。
mojikumi	string	The Mojikumi name.
parent	object	只读。对象的容器。
rightIndent	number (double)	右边距缩进（以磅为单位）。
romanHanging	boolean	如果为true，则启用罗马悬挂标点符号。
singleWordJustification	Justification	单字对齐。
spaceAfter	number (double)	段落后的间距（以磅为单位）。
spaceBefore	number (double)	段落前的间距（以磅为单位）。
tabStops	TabStopInfo	制表位设置。
typename	string	只读。对象的类名称。

Changing justification in paragraphs

// 创建一个具有1个文本框架和3个段落的新文档然后为每个段落指定不同的对齐方式

```
var docRef = documents.add();
var pathRef = docRef.pathItems.rectangle(600, 200, 200, 400);
var 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;
```

// 更改每个段落的对齐样式

// 使用段落属性对象

```
var paraAttr_0 = textRef.paragraphs[0].paragraphAttributes;
paraAttr_0.justification = Justification.RIGHT;
var paraAttr_1 = textRef.paragraphs[1].paragraphAttributes;
paraAttr_1.justification = Justification.CENTER;
var paraAttr_2 = textRef.paragraphs[2].paragraphAttributes;
paraAttr_2.justification = Justification.LEFT;
```

Paragraphs

`TextRange`对象的集合，每个`TextRange`代表一个段落。元素未命名；您必须按索引访问它们。

Paragraphs properties

属性	值类型	注释
<code>length</code>	number	只读。集合中的对象数。
<code>parent</code>	object	只读。此对象的父级。
<code>typename</code>	string	只读。被引用对象的类名。

Paragraphs methods

方法	参数类型	返回值	作用
<code>add</code> (contents [,relativeObject] [,insertionLocation])	string TextFrameItem ElementPlacement	TextRange	在当前文档中的指定位置添加具有指定文本内容的新段落。如果未指定 <code>location</code> ，则将新段落添加到当前文本选择或插入点之后的包含文本框中。
<code>addBefore</code> (contents)	string	TextRange	Adds a new paragraph with specified text contents before the current text selection or insertion point.
<code>index</code> (itemKey)	number	TextRange	从集合中获取一个元素。
<code>removeAll</code> ()		Nothing	删除此集合中的所有元素。

段落计数

```
//计算当前文档中的所有段落并将结果存储在paragraphCount中
if ( app.documents.length > 0 ) {
  doc = app.activeDocument;
  paragraphCount = 0;
  for ( i = 0; i < doc.textFrames.length; i++ ) {
    paragraphCount += doc.textFrames[i].paragraphs.length;
  }
}
```

ParagraphStyle

Associates character and paragraph attributes with a style name. The style object can be used to apply those attributes to the text in a `TextFrame` object. See [Creating and applying a paragraph style](#) below.

ParagraphStyle properties

属性	值类型	注释
<code>characterAttributes</code>	CharacterAttributes	只读。文本范围的字符属性。
<code>name</code>	string	段落样式的名称。
<code>paragraphAttributes</code>	ParagraphAttributes	只读。文本范围的段落属性。
<code>parent</code>	object	只读。对象的容器。
<code>typename</code>	string	只读。对象的类名称。

ParagraphStyle methods

方法	参数类型	返回值	作用
<code>applyTo</code> (textItem [,clearingOverrides])	object boolean	Nothing	Applies this paragraph style to the specified text item.
<code>remove</code> ()		Nothing	删除对象。

ParagraphStyles

A collection of `ParagraphStyle` objects.

ParagraphStyles properties

属性	值类型	注释
<code>length</code>	<code>number</code>	只读。集合中元素的数量。
<code>parent</code>	<code>object</code>	只读。对象的容器。
<code>typename</code>	<code>string</code>	只读。对象的类名称。

ParagraphStyles methods

方法	参数类型	返回值	作用
<code>add</code> (name)	<code>string</code>	ParagraphStyle	创建命名的段落样式。
<code>getByName</code> (name)	<code>string</code>	ParagraphStyle	使用提供的名称获取集合中的第一个元素。
<code>index</code> (itemKey)	<code>string, number</code>	ParagraphStyle	从集合中获取一个元素。
<code>removeAll</code> ()		Nothing	删除集合中的所有元素。

创建和应用段落样式

```
//创建一个带有1个文本框架和3个段落的新文档
//为每个段落提供不同的对齐方式，然后创建
//段落样式并将其应用于所有段落

var docRef = documents.add();
var pathRef = docRef.pathItems.rectangle(600, 200, 200, 400);
var 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;

//使用段落属性对象更改每个段落的对齐方式
var paraAttr_0 = textRef.paragraphs[0].paragraphAttributes;
paraAttr_0.justification = Justification.RIGHT;
var paraAttr_1 = textRef.paragraphs[1].paragraphAttributes;
paraAttr_1.justification = Justification.CENTER;
var paraAttr_2 = textRef.paragraphs[2].paragraphAttributes;
paraAttr_2.justification = Justification.LEFT;

//创建新的段落样式
var paraStyle = docRef.paragraphStyles.add("LeftIndent");

//添加一些段落属性
var paraAttr = paraStyle.paragraphAttributes;
paraAttr.justification = Justification.LEFT;
paraAttr.firstLineIndent = 10;

//将样式应用于文档中的每个项目
var iCount = textRef.paragraphs.length;
for(var i=0; i<iCount; i++) {

    paraStyle.applyTo(textRef.paragraphs[i], true);
}
redraw();
```

PathItem

指定一个路径项，其中包含定义其几何形状的PathPoint对象。PathItem类使您可以完全访问Illustrator中的路径。setEntirePath方法提供了一种非常有效的方法来创建由直线组成的路径。

PathItem properties

属性	值类型	注释
area	number (double)	只读。该路径的面积（以平方点为单位）。如果该区域为负，则该路径将逆时针缠绕。自相交路径可以包含相互抵消的子区域，即使该路径具有明显的面积，该子区域也使该值变为零。
artworkKnockout	KnockoutState	此对象用于创建蒙版对象吗？如果是，则使用哪种蒙版类型。
blendingMode	BlendModes	合成对象时使用的混合模式。
clipping	boolean	如果为true，则应将此路径用作剪切路径。
closed	boolean	如果为true，则关闭此路径。
controlBounds	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
editable	boolean	只读。如果为true，则此项目是可编辑的。
evenodd	boolean	如果为true，则应使用奇偶规则确定“insideness。”
fillColor	Color	路径的填充颜色。
filled	boolean	如果为true，则填充路径。
fillOverprint	boolean	如果为true，则应套印填充对象下的艺术品。
geometricBounds	array of 4 numbers	只读。不包含笔划宽度的对象边界。
guides	boolean	如果为true，则此路径为参考线。
height	number (double)	组项目的高度。
hidden	boolean	如果为true，则此项是隐藏的。
isIsolated	boolean	如果为true，则隔离此对象。
layer	Layer	只读。此项所属的图层。
left	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。

属性	值类型	注释
length	number (double)	该路径的长度 (以磅为单位)。
locked	boolean	如果为true, 则该项被锁定。
name	string	此项的名称。
note	string	分配给路径的注释文本。
opacity	number (double)	对象的不透明度。范围: 0.0至100.0
parent	Layer Or GroupItem	只读。此对象的父级。
pathPoints	PathPoints	只读。此路径项中包含的路径点。
pixelAligned	boolean	如果此项与像素网格对齐, 则为True。
polarity	PolarityValues	路径的极性。
position	array of 2 numbers	pathItem 对象左上角的位置 (以磅为单位), 格式为[x, y]。不包括行程重量。
resolution	number (double)	路径的分辨率, 以每英寸点数 (dpi) 为单位。
selected	boolean	如果为true, 则选择此项目。
selectedPathPoints	PathPoints	只读。路径中所有选定的路径点。
sliced	boolean	如果为true, 则将项目选中。默认值: false
strokeCap	StrokeCap	线路封顶的类型。
strokeColor	Color	路径的笔触颜色。
stroked	boolean	如果为true, 则应绘制路径。
strokeDashes	object	短划线长度。设置为空对象 {}, 以获得实线。
strokeDashOffset	number (double)	距虚线图案的默认距离, 应从该距离开始。
strokeJoin	StrokeJoin	路径的关节类型。
strokeMiterLimit	number (double)	当默认笔触连接设置为“斜接”时, 此属性指定默认情况下何时将连接转换为斜角 (平方)。默认的斜接限制为4表示当点的长度达到笔划粗细的四倍时, 连接将从斜接连接切换为斜角连接。值1指定一个斜角连接。范围: 1到500。默认值: 4
strokeOverprint	boolean	如果为true, 则应套印描边对象下方的艺术品。
strokeWidth	number (double)	笔划的宽度 (以磅为单位)。

属性	值类型	注释
tags	Tags	只读。此项目中包含的标签。
top	number (double)	项目顶部的位置（以磅为单位，从页面底部开始测量）。
typename	string	只读。被引用对象的类名。
uRL	string	分配给此项的Adobe URL标记的值。
visibilityVariable	Variable	绑定到项目的可见性变量。
visibleBounds	array of 4 numbers	只读。项目的可见边界，包括描边宽度。
width	number (double)	项目的宽度。
wrapInside	boolean	如果为true，则应将文本框架对象包装在该对象内。
wrapOffset	number (double)	在此对象上环绕文字时要使用的偏移量。
wrapped	boolean	如果为true，则将文本框架对象包装在该对象周围（文本框架必须在该对象上方）。
zOrderPosition	number (long)	只读。该项目在包含该项目的组或层（父级）的堆叠顺序中的位置。

PathItem methods

方法	参数类型	返回值	作用
duplicate ([relativeObject] [, insertionLocation])	object ElementPlacement	PathItem	创建所选对象的副本。
move (relativeObject, insertionLocation)	object ElementPlacement	PathItem	移动对象。
remove ()		Nothing	删除该对象。
resize (scaleX, scaleY [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, scaleAbout])	number (double) number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	缩放艺术品，其中scaleX是水平缩放因子，而scaleY是垂直缩放因子。100.0 = 100%。

方法	参数类型	返回值	作用
rotate (angle [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, rotateAbout])	number (double) boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转艺术品。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象。
setEntirePath (pathPoints)	array of [x, y] coordinate pairs	Nothing	使用指定为[x, y]坐标对的点数组来设置路径。
transform (transformationMatrix [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, transformAbout])	Matrix boolean boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换艺术品。
translate ([deltaX] [, deltaY] [, transformObjects] [, transformFillPatterns] [, transformFillGradients] [, transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新放置艺术品，其中delX是水平偏移，而delY是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列艺术品的位置。

在路径中设置颜色

// 将路径项目的笔触和填充设置为随机选择的色板的颜色

```

if ( app.documents.length > 0 && app.activeDocument.pathItems.length > 0 ) {
  doc = app.activeDocument;
  for (var i = 0; i < doc.pathItems.length; i++ ) {
    pathRef = doc.pathItems[i];
    pathRef.filled = true;
    pathRef.stroked = true;
    swatchIndex = Math.round( Math.random() * ( doc.swatches.length - 1 ) );
    pathRef.fillColor = doc.swatches[ swatchIndex ].color;
    pathRef.strokeColor = doc.swatches[ swatchIndex ].color;
  }
}

```

从直线创建路径

该脚本说明了setEntirePath方法的用法。

// 创建由10条直线组成的新开放路径

```
if ( app.documents.length > 0 ) {  
    var lineList = new Array(10);  
    for ( i = 0; i < lineList.length; i++ ) {  
        lineList[i] = new Array( i * 10 + 50, ((i - 5) ^ 2) * 5 + 50);  
    }  
    app.defaultStroked = true;  
    newPath = app.activeDocument.pathItems.add();  
    newPath.setEntirePath(lineList);  
}
```

PathItems

`PathItem`对象的集合。椭圆，多边形，矩形，`roundedRectangle`和`star`方法允许您使用简单的参数创建复杂的路径项。如果在调用这些方法时未提供任何参数，则使用默认值。

PathItems properties

属性	值类型	注释
<code>length</code>	number	只读。集合中的对象数。
<code>parent</code>	object	只读。此对象的父级。
<code>typename</code>	string	只读。被引用对象的类名。

PathItems methods

方法	参数类型	返回值	作用
<code>add</code> ()		PathItem	创建一个新对象。
<code>ellipse</code> ([top] [, left] [, width] [, height] [, reversed] [, inscribed])	number (double) number (double) number (double) number (double) boolean boolean	PathItem	使用提供的参数创建一个椭圆形的新 <code>pathItem</code> 。默认值：顶部：100磅；左：100点；宽度：50点；高度：100磅；反向：错误
<code>getByName</code> (name)	string	PathItem	获取具有指定名称的集合中的第一个元素。
<code>index</code> (itemKey)	string, number	PathItem	从集合中获取一个元素。
<code>polygon</code> ([centerX] [, centerY] [, radius] [, sides] [, reversed])	number (double) number (double) number (double) number (long) boolean	PathItem	使用提供的参数创建多边形形状的新 <code>pathItem</code> 。默认值：centerX: 200 pt .; 中心Y: 300点；半径：50点；边：8；反向：错误
<code>rectangle</code> (top, left, width, height [, reversed])	number (double) number (double) number (double) number (double) boolean	PathItem	使用提供的参数创建多边形形状的新 <code>pathItem</code> 。
<code>removeAll</code> ()		Nothing	删除此集合中的所有元素。

方法	参数类型	返回值	作用
roundedRectangle (top, left, width, height [,horizontalRadius] [,verticalRadius] [,reversed])	number (double) number (double) number (double) number (double) number (double) number (double) boolean	PathItem	使用提供的参数创建一个带有圆角的矩形形状的新pathItem。默认值: horizontalRadius: 15 pt; verticalRadius: 20pt。 反向: 错误
star ([centerX] [,centerY] [,radius] [,innerRadius] [,points] [,reversed])	number (double) number (double) number (double) number (double) number (long) boolean	PathItem	使用提供的参数创建星形的新路径项。默认值: centerX: 200pt; 中心Y: 300pt; 半径: 50pt; 内部半径: 20pt; 分: 5; 反向: 错误

创建形状

```
//在文档1的第1层中创建5个形状
//并对每个形状应用随机的图形样式

var doc = app.documents.add();
var artLayer = doc.layers[0];
app.defaultStroked = true;
app.defaultFilled = true;

var rect = artLayer.pathItems.rectangle( 762.5, 87.5, 425.0, 75.0 );
var rndRect = artLayer.pathItems.roundedRectangle(
    637.5, 87.5, 425.0, 75.0, 20.0, 10.0 );

//创建椭圆, “reversed”为false, “inscribed”为true
var ellipse = artLayer.pathItems.ellipse(
    512.5, 87.5, 425.0, 75.0, false, true );

//创建八边形和8面多边形
var octagon = artLayer.pathItems.polygon( 300.0, 325.0, 75.0, 8 ); //
Create a 4 pointed star
var star = artLayer.pathItems.star( 300.0, 125.0, 100.0, 20.0, 4 );

for ( i = 0; i < artLayer.pathItems.length; i++ ) {
    styleIndex = Math.round(
        Math.random() * ( doc.graphicStyles.length - 1 ) );
    doc.graphicStyles[styleIndex].applyTo( artLayer.pathItems[i] );
}
```


PathPoint

特定路径上的一点。每个路径点都由一个锚点（锚点）和一对手柄（`leftDirection`和`rightDirection`）组成。

PathPoint properties

属性	值类型	注释
<code>anchor</code>	array of 2 numbers	该点的锚点的位置。
<code>leftDirection</code>	array of 2 numbers	此路径点的位置在控制点中。
<code>parent</code>	PathItem	只读。包含此路径点的路径项。
<code>pointType</code>	PointType	路径点的类型，可以是曲线或拐角。任何点都可以视为角点。将类型设置为角时，当用户尝试在用户界面中修改左右方向点时，它们将强制位于一条直线上。
<code>rightDirection</code>	array of 2 numbers	此路径点位于输出控制点的位置。
<code>selected</code>	PathPointSelection	是否选择了该路径点的点，如果已选择，则选择了哪些点。
<code>typename</code>	string	只读。被引用对象的类名。

PathPoint methods

方法	参数类型	返回值	作用
<code>remove()</code>		Nothing	从路径中删除参考点。

PathPoints

特定路径中PathPoint对象的集合。元素未命名；您必须按索引访问它们。

PathPoints properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

PathPoints methods

方法	参数类型	返回值	作用
add ()		PathPoint	创建一个新的PathPoint对象。
index (itemKey)	number	PathPoint	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

向路径添加路径点

//将新的PathPoint追加到现有路径
//并初始化其锚点和句柄点。

```
if ( app.documents.length > 0 ) {
    var doc = app.activeDocument;
    var line = doc.pathItems.add();
    line.stroked = true;
    line.setEntirePath( Array( Array(220, 475), Array(375, 300) ) );

    // 将另一点附加到该行
    var newPoint = doc.pathItems[0].pathPoints.add();

    newPoint.anchor = Array(220, 300);
    newPoint.leftDirection = newPoint.anchor;
    newPoint.rightDirection = newPoint.anchor;
    newPoint.pointType = PointType.CORNER;
}
```

Pattern

文档中包含的Illustrator模式定义。模式显示在“色板”调板中。每个图案都由 [PatternColor](#) 对象引用，该对象定义了图案的外观。

Pattern properties

属性	值类型	注释
name	string	模式名称。
parent	Document	只读。包含此模式的文档。
typename	string	只读。被引用对象的类名。

Pattern methods

方法	参数类型	返回值	作用
remove ()		Nothing	从文档中删除引用的模式。
toString ()		string	返回引用对象的对象类型。如果对象具有名称，则还返回名称。

PatternColor

图案颜色规范。您可以通过修改文档中的现有图案来创建新的图案颜色。您对图案进行的任何修改都会影响组件面板中的该图案。

PatternColor对象可以在带有颜色对象的任何属性中使用，例如fillColor或strokeColor。

PatternColor properties

属性	值类型	注释
matrix	Matrix	通过操纵路径而产生的其他转换。
pattern	Pattern	对图案对象的引用，该对象定义要在此颜色定义中使用的图案。
reflect	boolean	如果为true，则应在填充之前反映原型。默认值: false
reflectAngle	number (double)	反射所围绕的轴（以磅为单位）。默认值: 0.0
rotation	number (double)	以弧度为单位的角度，以在填充前旋转原型图案。默认值: 0.0
scaleFactor	array of 2 numbers	填充之前缩放原型图案的比例，表示为包含水平和垂直缩放百分比的点。
shearAngle	number (double)	倾斜剪切的角度（以弧度为单位）。默认值: 0.0
shearAxis	number (double)	相对于点的剪切轴。默认值: 0.0
shiftAngle	number (double)	填充前将未缩放的原型图案转换为的弧度角。默认值: 0.0
shiftDistance	number (double)	填充前将未缩放的原型图案转换为的距离，以点为单位。默认值: 0.0
typename	string	只读。被引用对象的类名。

修改和应用图案颜色

```
//旋转当前文档中每个图案的颜色，
//然后将最后一个图案应用于第一个路径项

if ( app.documents.length > 0 && app.activeDocument.pathItems.length > 0 ) {
    doc = app.activeDocument;
    swatchIndex = 0;
    for ( i = 0; i < doc.swatches.length; i++ ) {
        //获取样例的通用颜色对象 currentSwatch = doc.swatches[i];
        swatchColor = currentSwatch.color;
        //仅对模式进行操作
        if ( currentSwatch.color.typename == "PatternColor" ) {

            //更改模式属性
            currentSwatch.color.rotation = 10;
            swatchIndex = i;
        }
    }
    //将最后一个图案色样应用于最前面的路径
    firstPath = app.activeDocument.pathItems[0];
    firstPath.filled = true;
    firstPath.fillColor = doc.swatches[swatchIndex].color;
}
```

Patterns

文档中Pattern对象的集合。

Patterns properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

Patterns methods

方法	参数类型	返回值	作用
add ()		Pattern	创建一个新对象。
getName (name)	string	Pattern	使用提供的名称获取集合中的第一个元素。
index (itemKey)	string, number	Pattern	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

删除图案

//从当前文档中删除最后一个模式

```
if ( app.documents.length > 0 ) {
    var lastIndex = app.activeDocument.patterns.length - 1;
    var patternToRemove = app.activeDocument.patterns[lastIndex];
    var patternName = patternToRemove.name;
    patternToRemove.remove();
    //注意在删除Illustrator对象之后，将
    //引用删除的对象的变量设置为null，因为它现在无效。
    patternToRemove = null;
}
```

PDFFileOptions

用于打开PDF文件的选项，与 [open](#) 方法一起使用。所有属性都是可选的。

PDFFileOptions properties

属性	值类型	注释
<code>pageToOpen</code>	number (long)	打开多页文档时应使用哪一页。 默认: 1
<code>parent</code>	object	只读。对象的容器。
<code>pdfCropToBox</code>	PDFBoxType	放置多页文档时应使用哪个框。默认值 (媒体框) : <code>PDFBoxType.PDFMediaBox</code>
<code>typename</code>	string	只读。对象的类名称。

使用选项打开PDF

```
//打开具有指定选项的PDF文件

var pdfOptions = app.preferences.PDFFileOptions;
pdfOptions.pdfCropToBox = PDFBoxType.PDFBOUNDINGBOX;
pdfOptions.pageToOpen = 2;

//使用这些首选项打开文件
var fileRef = filePath;
if (fileRef != null) {
    var docRef = open(fileRef, DocumentColorSpace.RGB);
}
```

PDFSaveOptions

用于将文档另存为Adobe PDF文件的选项，与 [saveAs](#) 方法一起使用。所有属性都是可选的。

PDFSaveOptions properties

属性	值类型	注释
<code>acrobatLayers</code>	boolean	可选的。从顶层创建 Acrobat®层。仅限于 Acrobat 6。默认： <code>false</code>
<code>artboardRange</code>	string	可选的。考虑使用多资产提取来指定画板范围。空字符串将提取所有画板。默认值：空字符串
<code>bleedLink</code>	boolean	可选的。链接4个出血值。默认值： <code>true</code>
<code>bleedOffsetRect</code>	array of 4 numbers	出血偏移矩形。
<code>colorBars</code>	boolean	可选的。绘制颜色条。默认值： <code>false</code>
<code>colorCompression</code>	CompressionQuality	可选的。使用的颜色位图压缩的类型。默认： <code>CompressionQuality.None</code>
<code>colorConversionID</code>	ColorConversion	可选的。PDF颜色转换策略。默认： <code>ColorConversion.None</code>
<code>colorDestinationID</code>	ColorDestination	可选的。颜色转换的转换目标。默认： <code>ColorDestination.None</code>
<code>colorDownsampling</code>	number (double)	可选的。颜色下采样分辨率，以每英寸点数 (dpi) 为单位。如果为0，则不执行下采样。默认值： <code>150.0</code>
<code>colorDownsamplingImageThreshold</code>	number (double)	可选的。如果图像的分辨率高于该值，则降低采样率。默认值： <code>225.0</code>

属性	值类型	注释
<code>colorDownsamplingMethod</code>	DownsampleMethod	可选的。彩色位图图像应如何重新采样。默认： <code>DownsampleMethod.NODOWNSAMPLE</code>
<code>colorProfileID</code>	ColorProfile	可选的。要包括的颜色配置文件。默认： <code>ColorProfile.None</code>
<code>colorTileSize</code>	number (long)	可选的。使用 JPEG2000 压缩时的图块大小。默认值： 256
<code>compatibility</code>	PDFCompatibility	可选的。要创建的Acrobat文件格式的版本。默认： <code>PDFCompatibility.Acrobat5</code>
<code>compressArt</code>	boolean	可选的。如果为true，则应压缩线条和文字。 默认值： <code>true</code>
<code>documentPassword</code>	string	可选的。打开文档的密码字符串。默认值：无字符串
<code>enableAccess</code>	boolean	可选的。如果为true，则启用访问128位。默认值： <code>true</code>
<code>enableCopy</code>	boolean	可选的。如果为true，则启用复制128位文本。默认值： <code>true</code>
<code>enableCopyAccess</code>	boolean	可选的。如果为true，则启用复制和访问40位。默认值： <code>true</code>
<code>enablePlainText</code>	boolean	可选的。如果为true，则启用128位纯文本元数据。仅适用于Acrobat6。默认值： <code>false</code>
<code>flattenerOptions</code>	PrintFlattenerOptions	可选的。打印展平器选项。
<code>flattenerPreset</code>	stringOptional.	可选的。透明度展平器预设名称。

属性	值类型	注释
<code>fontSubsetThreshold</code>	number (double)	可选的。当文档中使用的字符数少于此百分比时，请包括字体的子集。对 Illustrator 9 文件格式有效。范围：0.0到100.0。默认值：100.0
<code>generateThumbnails</code>	boolean	可选的。如果为true，则将使用保存的文件生成缩略图。默认值：true
<code>grayscaleCompression</code>	CompressionQuality	可选的。灰度位图压缩的质量。默认值：无
<code>grayscaleDownsampling</code>	number (double)	可选的。下采样分辨率，以每英寸点数 (dpi) 为单位。如果为0，则不执行下采样。默认值：150.0
<code>grayscaleDownsamplingImageThreshold</code>	number (double)	可选的。如果图像的分辨率高于该值，则降低采样率。默认值：225.0
<code>grayscaleDownsamplingMethod</code>	DownsampleMethod	可选的。灰度位图图像应如何重新采样默认值：DownSampleMethod.NODOWNSAMPLE
<code>grayscaleTileSize</code>	number (long)	可选的。使用 JPEG2000 压缩时的图块大小。默认值：256
<code>monochromeCompression</code>	MonochromeCompression	可选的。使用的单色位图压缩类型。默认： MonochromeCompression.None
<code>monochromeDownsampling</code>	number (double)	可选的。下采样分辨率，以每英寸点数 (dpi) 为单位。如果为0，则不执行下采样。默认值：300
<code>monochromeDownsamplingImageThreshold</code>	number (double)	可选的。如果图像的分辨率高于该值，则降低采样率。默认值：450.0

属性	值类型	注释
<code>monochromeDownsamplingMethod</code>	DownsampleMethod	可选的。单色位图图像应如何重新采样。默认： <code>DownSampleMethod.NODOWNSAMPLE</code>
<code>offset</code>	number (double)	可选的。自定义偏移量（以磅为单位）以使用自定义纸张。默认值： <code>0.0</code>
<code>optimization</code>	boolean	可选的。如果为 <code>true</code> ，则应优化PDF文档以快速浏览Web。默认值： <code>false</code>
<code>outputCondition</code>	string	可选的。添加到PDF文件的可选注释，描述了预期的打印条件。默认值：不包括
<code>outputConditionID</code>	string	可选的。注册的打印条件的名称。默认值：不包括
<code>pageInformation</code>	boolean	可选的。如果为 <code>true</code> ，则为原始页面信息。默认值： <code>false</code>
<code>pageMarksType</code>	PageMarksTypes	可选的。页面标记样式。默认： <code>PageMarksType.Roman</code>
<code>pdfAllowPrinting</code>	PDFPrintAllowedEnum	可选的。PDF安全打印许可。默认： <code>PDFPrintAllowedEnum.PRINT128HIGHRESOLUTION</code>
<code>pdfChangesAllowed</code>	PDFChangesAllowedEnum	可选的。允许更改安全性。默认： <code>PDFChangeAllowedEnum.CHANGE128ANYCHANGES</code>
<code>pdfPreset</code>	string	可选的。要使用的PDF预设名称。
<code>pdfXStandard</code>	PDFXStandard	可选的。本文档所遵循的PDF标准。默认： <code>PDFXStandard.PDFXNONE</code>
<code>pdfXStandardDescription</code>	string	可选的。所选预设中PDF标准的描述。

属性	值类型	注释
<code>permissionPassword</code>	string	可选的。密码字符串，用于限制编辑安全设置。默认值：无字符串
<code>preserveEditability</code>	boolean	可选的。如果为true，则在保存文档时应保留Illustrator编辑功能。默认值：true
<code>printerResolution</code>	number (double)	可选的。展平打印机分辨率。默认值：800.0
<code>registrationMarks</code>	boolean	可选的。如果为true，请绘制注册标记。默认值：false
<code>requireDocumentPassword</code>	boolean	可选的。需要密码才能打开文档。默认值：false
<code>requirePermissionPassword</code>	boolean	可选的。使用密码来限制编辑安全性设置。默认值：false
<code>trapped</code>	boolean	可选的。如果为true，则已经为文档准备了手动陷印。默认值：false
<code>trimMarks</code>	boolean	可选的。绘制修剪标记。默认值：false
<code>trimMarkWeight</code>	PDFTrimMarkWeight	可选的。修剪标记的重量。默认： PDFTrimMarkWeight. TRIMMARKWEIGHT0125
<code>typename</code>	string	Optional. 只读。被引用对象的类名。
<code>viewAfterSaving</code>	boolean	可选的。保存后查看PDF。默认值：false

保存为PDF格式

```
//使用指定的选项将当前文档另存为PDF到dest
// dest包含要保存到的完整路径和文件名

function saveFileToPDF (dest) {
  var doc = app.activeDocument;
  if ( app.documents.length > 0 ) {
    var saveName = new File ( dest );
    saveOpts = new PDFSaveOptions();
    saveOpts.compatibility = PDFCompatibility.ACROBAT5;
    saveOpts.generateThumbnails = true;
    saveOpts.preserveEditability = true;
    doc.saveAs( saveName, saveOpts );
  }
}
```

PhotoshopFileOptions

用于打开Photoshop文件的选项，与 [open](#) 方法一起使用。所有属性都是可选的。

PhotoshopFileOptions properties

属性	值类型	注释
<code>parent</code>	<code>object</code>	只读。此对象的父级。
<code>pixelAspectRatioCorrection</code>	<code>boolean</code>	如果为 <code>true</code> ，则应调整具有非正方形像素长宽比的导入图像。
<code>preserveImageMaps</code>	<code>boolean</code>	如果为 <code>true</code> ，则在转换文档时应保留图像映射。默认值： <code>true</code>
<code>preserveLayers</code>	<code>boolean</code>	如果为 <code>true</code> ，则转换文档时应保留图层。默认值： <code>true</code>
<code>preserveSlices</code>	<code>boolean</code>	如果为 <code>true</code> ，则在转换文档时应保留切片。默认值： <code>true</code>
<code>typename</code>	<code>string</code>	只读。被引用对象的类名。

打开一个Photoshop文件

```
//打开一个Photoshop文件，其中包含设置了  
//首选项以保留图层的图层
```

```
var psdOptions = preferences.photoshopFileOptions;  
psdOptions.preserveLayers = true;  
psdOptions.pixelAspectRatioCorrection = false;  
//使用这些首选项打开文件  
var fileRef = File( psdFilePath);  
if (fileRef != null) {  
    var docRef = open(fileRef, DocumentColorSpace.RGB);  
}
```

PlacedItem

作为链接文件放置在文档中的艺术品。例如，使用Illustrator中的“文件”>“置入”命令或使用placedItems集合对象的add () 方法创建的艺术品对象是放置项。有关信息，请参阅第155页上的[“PlacedItems” on page 155](#)。

PlacedItem properties

属性	值类型	注释
artworkKnockout	KnockoutState	此对象用于创建蒙版对象吗？如果是，则使用哪种蒙版类型。
blendingMode	BlendModes	合成对象时使用的混合模式。
boundingBox	array of 4 numbers	只读。放置的艺术品的尺寸，与变换无关。
contentVariable	Variable	绑定到项目的content变量。
controlBounds	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
editable	boolean	只读。如果为true，则此项目是可编辑的。
file	File	包含插图的文件。
geometricBounds	array of 4 numbers	只读。除笔划宽度以外的对象边界。
height	number (double)	组项目的高度。
hidden	boolean	如果为true，则此项是隐藏的。
isIsolated	boolean	如果为true，则隔离此对象。
layer	Layer	只读。此项所属的图层。
left	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。
locked	boolean	如果为true，则该项被锁定。
matrix	Matrix	放置的艺术品的转换矩阵。
name	string	此项的名称。
note	string	该项目的注释。
opacity	number (double)	对象的不透明度。范围：0.0至100.0
parent	Layer or GroupItem	只读。此对象的父级。
position	array of 2 numbers	以[x, y]格式表示placedItem对象左上角的位置（以磅为单位）。不包括行程重量。
selected	boolean	如果为true，则选择此项目。

属性	值类型	注释
sliced	boolean	如果为true, 则将项目选中。默认值: false
tags	Tags	只读。此项目中包含的标签。
top	number (double)	项目顶部的位置 (以磅为单位, 从页面底部开始测量)。
typename	string	只读。被引用对象的类名。
uRL	string	分配给此项的Adobe URL标记的值。
visibilityVariable	Variable	绑定到项目的可见性变量。
visibleBounds	array of 4 numbers	只读。项目的可见边界, 包括描边宽度。
width	number (double)	项目的宽度。
wrapInside	boolean	如果为true, 则应将文本框架对象包装在该对象内。
wrapOffset	number (double)	在此对象上环绕文字时要使用的偏移量。
wrapped	boolean	如果为true, 则将文本框架对象包装在该对象周围 (文本框架必须在对象上方)。
zOrderPosition	number (long)	只读。该项目在包含该项目的组或层 (父级) 的堆叠顺序中的位置。

PlacedItem methods

方法	参数类型	返回值	作用
duplicate ([relativeObject] [,insertionLocation])	object ElementPlacement	PlacedItem	Creates a duplicate of the selected object.
embed ()		Nothing	将这种艺术嵌入文档中。根据需要将艺术品转换为艺术品对象, 并删除该对象。
move (relativeObject, insertionLocation)	object ElementPlacement	PlacedItem	移动对象。
relink (linkFile)	File object	Nothing	将艺术品与定义其内容的文件重新链接。
remove ()		Nothing	删除该对象。

方法	参数类型	返回值	作用
resize (scaleX, scaleY [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, scaleAbout])	number (double) number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	缩放艺术品，其中scaleX是水平缩放因子，而scaleY是垂直缩放因子。100.0 = 100%。
rotate (angle [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, rotateAbout])	number (double) boolean boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转艺术品。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象。
trace ()		PluginItem	Converts the raster art for this object to vector art, using default options. Reorders the placed art into the source art of a plug-in group, and converts it into a group of filled and/or stroked paths that resemble the original image. 创建并返回一个引用tracingObject对象的pluginItem对象。
transform (transformationMatrix [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, transformAbout])	Matrix boolean boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换艺术品。
translate ([deltaX] [, deltaY] [, transformObjects] [, transformFillPatterns] [, transformFillGradients] [, transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新定位艺术品，其中deltaX是水平偏移量，deltaY是垂直偏移量。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列艺术品的位置。

更改放置项目的选择状态

//切换所有放置项目的选择状态。

```
if ( app.documents.length > 0 ) {  
    for ( i = 0; i < app.activeDocument.placedItems.length; i++ ) {  
        placedArt = app.activeDocument.placedItems[i];  
        placedArt.selected = !(placedArt.selected);  
    }  
}
```

PlacedItems

文档中PlacedItem对象的集合。

PlacedItems properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

PlacedItems methods

方法	参数类型	返回值	作用
add ()	none	PlacedItem	创建一个新对象。 Use to place new art in a document. Use the <code>file</code> property of the resulting <code>placedItem</code> object to link the file containing the artwork. See “PlacedItem” on page 151 .
getByName (name)	string	PlacedItem	Gets the first element in the collection with the specified name.
index (itemKey)	string, number	PlacedItem	从集合中获取一个元素。
removeAll ()	none	Nothing	删除此集合中的所有元素。

PluginItem

由Illustrator插件创建的艺术品。脚本可以使用`PlacedItem.trace`或`RasterItem.trace`创建插件项目，并且可以使用重复方法复制现有插件项目，但是不能直接创建`PluginItem`对象。

PluginItem properties

属性	值类型	注释
<code>artworkKnockout</code>	KnockoutState	此对象用于创建蒙版对象吗？如果是，则使用哪种蒙版类型。
<code>blendingMode</code>	BlendModes	合成对象时使用的混合模式。
<code>controlBounds</code>	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
<code>editable</code>	boolean	只读。如果为 <code>true</code> ，则此项目是可编辑的。
<code>geometricBounds</code>	array of 4 numbers	只读。除笔划宽度以外的对象边界。
<code>height</code>	number (double)	组项目的高度。
<code>hidden</code>	boolean	如果为 <code>true</code> ，则此项是隐藏的。
<code>isIsolated</code>	boolean	如果为 <code>true</code> ，则隔离此对象。
<code>isTracing</code>	boolean	如果为 <code>true</code> ，则此插件组表示通过跟踪栅格艺术品而创建的矢量艺术品。跟踪属性包含与用于创建它的选项相关联的跟踪对象。
<code>layer</code>	Layer	只读。此项所属的图层。
<code>left</code>	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。
<code>locked</code>	boolean	如果为 <code>true</code> ，则该项被锁定。
<code>name</code>	string	此项的名称。
<code>note</code>	string	该项目的注释。
<code>opacity</code>	number (double)	对象的不透明度。范围：0.0至100.0
<code>parent</code>	Layer Or GroupItem	只读。此对象的父级。
<code>position</code>	array of 2 numbers	<code>pluginItem</code> 对象的左上角的位置（以磅为单位），格式为 <code>[x, y]</code> 。不包括行程重量。
<code>selected</code>	boolean	如果为 <code>true</code> ，则选择此项目。
<code>sliced</code>	boolean	如果为 <code>true</code> ，则将项目选中。默认值： <code>false</code>

属性	值类型	注释
tags	Tags	只读。此项目中包含的标签。
top	number (double)	项目顶部的位置（以磅为单位，从页面底部开始测量）。
tracing	TracingObject	通过跟踪创建此插件组（isTracing为true）时，与用于创建它的选项相关联的跟踪对象。
typename	string	只读。被引用对象的类名。
uRL	string	分配给此项的Adobe URL标记的值。
visibilityVariable	Variable	绑定到项目的可见性变量。
visibleBounds	array of 4 numbers	只读。项目的可见边界，包括描边宽度。
width	number (double)	项目的宽度。
wrapInside	boolean	如果为true，则应将文本框架对象包装在该对象内。
wrapOffset	number (double)	在此对象上环绕文字时要使用的偏移量。
wrapped	boolean	如果为true，则将文本框架对象包装在该对象周围（文本框架必须在对象上方）。
zOrderPosition	number	只读。该项目在包含该项目的组或层（父级）的堆叠顺序中的位置。

PluginItem methods

方法	参数类型	返回值	作用
duplicate ([relativeObject] [, insertionLocation])	object ElementPlacement	PluginItem	创建所选对象的副本。
move (relativeObject, insertionLocation)	object ElementPlacement	PluginItem	移动对象。
remove ()		Nothing	删除该对象。

方法	参数类型	返回值	作用
resize (scaleX, scaleY [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, scaleAbout])	number (double) number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	缩放艺术品，其中 scaleX 是水平缩放因子，而 scaleY 是垂直缩放因子。100.0 = 100%。
rotate (angle [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, rotateAbout])	number (double) boolean boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转艺术品。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象。
transform (transformationMatrix [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, transformAbout])	Matrix boolean boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换艺术品。
translate ([deltaX] [, deltaY] [, transformObjects] [, transformFillPatterns] [, transformFillGradients] [, transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新放置艺术品，其中 delX 是水平偏移，而 delY 是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列艺术品的位置。

复制插件项

//通过复制现有的插件艺术品来创建新的插件艺术品

```

if ( app.documents.length > 0 && app.activeDocument.pluginItems.length > 0 ) {
  doc = app.activeDocument;
  pluginArt = doc.pluginItems[0];
  pluginArt.duplicate( pluginArt.parent,
    ElementPlacement.PLACEATBEGINNING );
}

```

PluginItems

文档中PluginItem对象的集合。参见 [Copying a plug-in item](#)。

PluginItems properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

PluginItems methods

方法	参数类型	返回值	作用
getByName (name)	string	PluginItem	获取集合中的具有指定名称的一个元素
index (itemKey)	string, number	PluginItem	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有对象。

PPDFile

将文件信息与PostScript打印机描述（PPD）文件关联。

PPDFile properties

属性	值类型	注释
name	string	PPD型号名称。
PPDInfo	PPDFileInfo	PPD文件信息。
typename	string	只读。对象的类名称。

PPDFileInfo

有关PostScript打印机描述（PPD）文件的信息。

PPDFileInfo properties

属性	值类型	注释
<code>languageLevel</code>	string	PostScript语言级别。
<code>PPDFilePath</code>	File	PPD文件的路径规范。
<code>screenList</code>	array of Screen	分色屏幕列表。
<code>screenSpotFunctionList</code>	array of ScreenSpotFunction	分色屏幕专色功能列表。

显示PPD文件属性

//显示在新文本框中找到的每个PPD文件的脚本级别和路径

```

var sPPD = "";
var docRef = documents.add();
var x = 30;
var y = (docRef.height - 30);

var iLength = PPDFileList.length;
if (iLength > 20)
    iLength = 20;

for(var i=0; i<iLength; i++) {
    var ppdRef = PPDFileList[i];
    sPPD = ppdRef.name;
    sPPD += "\r\tPS Level ";
    var ppdInfoRef = ppdRef.PPDInfo;
    sPPD += ppdInfoRef.languageLevel;
    sPPD += "\r\tPath: ";
    sPPD += ppdInfoRef.PPDFilePath;

    var textRef = docRef.textFrames.add();
    textRef.textRange.characterAttributes.size = 8;
    textRef.contents = sPPD;
    textRef.top = (y);
    textRef.left = x;
    redraw();

    if( (y=(textRef.height)) <= 30 ) {
        y = (docRef.height - 30);
        x += 150;
    }
}

```

PPDFileInfo和相关屏幕信息

// 在新的文本框中显示后找到的10个PPD文件的脚本级别，文件路径，屏幕和屏幕位置信息

```

var sPPD = "";
var docRef = documents.add();
var x = 30;
var y = (docRef.height - 30);

var iLength = PPDFileInfo.length;
if (iLength > 10)
    iLength = 10;
for(var i=0; i<iLength; i++) {

    var ppdRef = PPDFileInfo[i];
    sPPD = ppdRef.name;
    sPPD += "\r\tPS Level ";
    var ppdInfoRef = ppdRef.PPDInfo;
    sPPD += ppdInfoRef.languageLevel;
    sPPD += "\r\tPath: ";
    sPPD += ppdInfoRef.PPDFilePath;

    sPPD += "\r\tScreens:\r";
    var iScreens = ppdInfoRef.screenList.length;
    for(var c=0; c<iScreens; c++) {
        var screenRef = ppdInfoRef.screenList[c];
        sPPD += "\t\t";
        sPPD += screenRef.name;
        var screenInfoRef = screenRef.screenInfo;
        sPPD += ", Angle = ";
        sPPD += screenInfoRef.angle;
        sPPD += ", Frequency = ";
        sPPD += screenInfoRef.frequency;
        sPPD += "\r";
    }

    sPPD += "\r\tScreenSpots:\r";
    var iScreenSpots = ppdInfoRef.screenSpotFunctionList.length;
    for(var n=0; n<iScreenSpots; n++) {
        var screenSpotRef = ppdInfoRef.screenSpotFunctionList[n];
        sPPD += "\t\t";
        sPPD += screenSpotRef.name;
        sPPD += ", spotFunction: ";
        sPPD += screenSpotRef.spotFunction;
        sPPD += "\r";
    }

    var textRef = docRef.textFrames.add();
    textRef.textRange.characterAttributes.size = 8;
    textRef.contents = sPPD;
    textRef.top = (y);
    textRef.left = x;
    redraw();

    y--(textRef.height);
}

```

Preferences

指定AutoCAD, FreeHand, PDF和Photoshop文件的首选选项。

Preferences properties

属性	值类型	注释
AutoCADFileOptions	OpenOptionsAutoCAD	只读。打开或放置AutoCAD文件时使用的选项。
FreeHandFileOptions	OpenOptionsFreeHand	只读。打开或放置FreeHand文件时使用的选项。
parent	object	只读。此对象的父级。
PDFFileOptions	PDFFileOptions	只读。打开或放置PDF文件时使用的选项。
PhotoshopFileOptions	PhotoshopFileOptions	只读。打开或放置Photoshop文件时使用的选项。
typename	string	只读。被引用对象的类名。

Preferences methods

方法	参数类型	返回值	作用
getBooleanPreference (key)	string	boolean	获取给定应用程序首选项的布尔值。
getIntegerPreference (key)	string	integer	获取给定应用程序首选项的整数值。
getRealPreference (key)	string	double	获取给定应用程序首选项的实数值。
getStringPreference (key)	string	string	获取给定应用程序首选项的字符串值。
removePreference (key)	string	Nothing	删除给定的应用程序首选项。
setBooleanPreference (key, value)	string boolean	Nothing	设置给定应用程序首选项的布尔值。
setIntegerPreference (key, value)	string integer	Nothing	设置给定应用程序首选项的整数值。

方法	参数类型	返回值	作用
setRealPreference (key, value)	string double	Nothing	设置给定应用程序首选项的实数值。
setStringPreference (key, value)	string string	Nothing	设置给定应用程序首选项的字符串值。

PrintColorManagementOptions

Information used for color management of the document.

PrintColorManagementOptions properties

属性	值类型	注释
colorProfileMode	PrintColorProfile	颜色管理配置文件模式。默认值： PrintColorProfile.SOURCEPROFILE
intent	PrintColorIntent	颜色管理意图类型。 默认值：PrintColorIntent.RELATIVECOLORIMETRIC
name	string	颜色管理配置文件名称。
typename	string	只读。对象的类名称。

管理打印颜色

```
//创建一个新文档，添加符号，然后创建一个
// PrintColorManagementOptions对象，并将其
//分配给一个PrintOptions对象，然后使用每种颜色进行打印

//将一些符号项添加到新文档
var docRef = documents.add();
var y = docRef.height - 30;
for (var i=0; i<(docRef.symbols.length); i++) {

    symbolRef = docRef.symbols[i];
    symbolItemRef1 = docRef.symbolItems.add(symbolRef);
    symbolItemRef1.top = y;
    symbolItemRef1.left = 100;
    y -= (symbolItemRef1.height + 10);
}
redraw();

var colorOptions = new PrintColorManagementOptions();
var options = new PrintOptions();
options.colorManagementOptions = colorOptions;
colorOptions.name = "ColorMatch RGB";

//为每种颜色意图打印一次当前文档。
colorOptions.intent = PrintColorIntent.ABSOLUTECOLORIMETRIC;
docRef.print(options);

colorOptions.intent = PrintColorIntent.PERCEPTUALINTENT;
docRef.print(options);

colorOptions.intent = PrintColorIntent.RELATIVECOLORIMETRIC;
docRef.print(options);

colorOptions.intent = PrintColorIntent.SATURATIONINTENT;
docRef.print(options);
```

PrintColorSeparationOptions

有关在打印文档中使用的分色的信息。

PrintColorSeparationOptions properties

属性	值类型	注释
<code>colorSeparationMode</code>	PrintColorSeparationMode	分色类型。默认： <code>PrintColorSeparationMode.COMPOSITE</code>
<code>convertSpotColors</code>	boolean	如果为true，则应将所有专色转换为原色。默认值： <code>false</code>
<code>inkList</code>	array of Ink	用于分色的墨水列表。
<code>overPrintBlack</code>	boolean	如果为true，则套印为黑色。默认值： <code>false</code>
<code>typename</code>	string	只读。对象的类名称。

管理打印分色

```
//创建带有符号项目的新文档
//并使用每个分隔选项打印文档

//将一些符号项添加到新文档
var docRef = documents.add();
var y = docRef.height - 30;
for(var i=0; i<(docRef.symbols.length); i++) {
    symbolRef = docRef.symbols[i];
    symbolItemRef1 = docRef.symbolItems.add(symbolRef);
    symbolItemRef1.top = y;
    symbolItemRef1.left = 100;
    y -= (symbolItemRef1.height + 10);
}
//使用各种分隔选项进行打印
var sepOptions = new PrintColorSeparationOptions();
var options = new PrintOptions();
options.colorSeparationOptions = sepOptions;

sepOptions.convertSpotColors = true;
sepOptions.overPrintBlack = true;
sepOptions.colorSeparationMode = PrintColorSeparationMode.COMPOSITE;
docRef.print(options);

sepOptions.colorSeparationMode = PrintColorSeparationMode.INRIPSEPARATION;
docRef.print(options);

sepOptions.convertSpotColors = false;
sepOptions.overPrintBlack = false;
sepOptions.colorSeparationMode = PrintColorSeparationMode.HOSTBASEDSEPARATION;
docRef.print(options);
```

PrintCoordinateOptions

有关介质和相关打印参数的信息。

PrintCoordinateOptions properties

属性	值类型	注释
emulsion	boolean	如果为true, 则水平翻转图稿。默认值: false
fitToPage	boolean	如果为true, 则按比例缩放图稿以适合媒体。默认值: false
horizontalScale	number (double)	水平缩放因子以百分比表示 (100 = 100%)。范围: 1.0至10000.0。默认值: 100.0
orientation	PrintOrientation	艺术品方向。 默认值: PrintOrientation.PORTRAIT
position	PrintPosition	艺术品在媒体上的位置。 默认值: PrintPosition.TRANSLATECENTER
tiling	PrintTiling	页面平铺模式。 默认值: PrintTiling.TILESINGLEFULLPAGE
typename	string	只读。对象的类名称。
verticalScale	number (double)	垂直比例因子以百分比表示 (100 = 100%) 范围: 1.0到10000.0。默认值: 100.0

管理打印坐标

//创建一个带有符号项的新文档, 这些符号项将从页面扩展
//然后以每个打印方向进行打印

```
var docRef = documents.add();
var y = 500;
var x = -70
if(docRef.symbols.length > 0){
  for(var i=0; i<5; i++) {
    symbolRef = docRef.symbols[0];
    symbolItemRef1 = docRef.symbolItems.add(symbolRef);
    symbolItemRef1.top = y;
    symbolItemRef1.left = x;
    x += 30;
  }
  redraw();
  //使用各种坐标选项进行打印
  var coordinateOptions = new PrintCoordinateOptions();
  var options = new PrintOptions();
  options.coordinateOptions = coordinateOptions;

  coordinateOptions.emulsion = true; //从右向左反转
  coordinateOptions.fitToPage = true; //使图稿适合页面大小
  coordinateOptions.orientation = PrintOrientation.LANDSCAPE;
```

```
docRef.print(options);
coordinateOptions.emulsion = false;
coordinateOptions.fitToPage = false;
coordinateOptions.orientation = PrintOrientation.PORTRAIT;
coordinateOptions.horizontalScale = 50;
coordinateOptions.verticalScale = 50;
docRef.print(options);
}
```


Printer

将可用的打印机与打印机信息相关联。要请求打印机列表，您必须首先打开文档或返回错误。

Printer properties

属性	值类型	注释
name	string	打印机名称。
printerInfo	PrinterInfo	打印机信息。
typename	string	只读。对象的类名称。

PrinterInfo

有关打印机的配置信息。

PrinterInfo properties

属性	值类型	注释
<code>binaryPrintingSupport</code>	boolean	如果为true, 则打印机支持二进制打印。
<code>colorSupport</code>	PrinterColorMode	打印机颜色功能。
<code>customPaperSupport</code>	boolean	如果为true, 则打印机支持自定义纸张尺寸。
<code>customPaperTransverseSupport</code>	boolean	如果为true, 则打印机支持自定义横向纸张。
<code>deviceResolution</code>	number (double)	打印机默认分辨率。
<code>inRIPSeparationSupport</code>	boolean	如果为true, 则打印机支持InRIP分色。
<code>maxDeviceResolution</code>	number (double)	打印机的最大设备分辨率。
<code>maxPaperHeight</code>	number (double)	自定义纸张的最大高度。
<code>maxPaperHeightOffset</code>	number (double)	自定义纸张的最大高度偏移量。
<code>maxPaperWidth</code>	number (double)	自定义纸张的最大宽度。
<code>maxPaperWidthOffset</code>	number (double)	自定义纸张的最大宽度偏移量。
<code>minPaperHeight</code>	number (double)	自定义纸张的最小高度。
<code>minPaperHeightOffset</code>	number (double)	自定义纸张的最小高度偏移量。
<code>minPaperWidth</code>	number (double)	自定义纸张的最小宽度。
<code>minPaperWidthOffset</code>	number (double)	自定义纸张的最小宽度偏移量。
<code>paperSizes</code>	array of Paper	支持的纸张尺寸列表。
<code>postScriptLevel</code>	PrinterPostScriptLevelEnum	PostScript语言级别。

属性	值类型	注释
<code>printerType</code>	PrinterTypeEnum	打印机类型。
<code>typename</code>	string	只读。对象的类名称。

查找可用的打印机

//在新文本框中显示可用打印机的列表

```
var docRef = documents.add();
var textRef = docRef.textFrames.add();

var iCount = printerList.length;
textRef.contents += "Printers...\r";
for( var i=0; i<iCount; i++ ) {
    textRef.contents += printerList[i].name;
    textRef.contents += "\r\t";
}
textRef.top = 600;
textRef.left = 200;
redraw();
```

PrintFlattenerOptions

包含展平选项，用于Illustrator将包含透明度的图稿输出为非本机格式时使用。

PrintFlattenerOptions properties

属性	值类型	注释
<code>clipComplexRegions</code>	boolean	如果为true，则应剪切复杂区域。默认值： false
<code>convertStrokesToOutlines</code>	boolean	如果为true，则将所有笔划转换为轮廓。默认值： false
<code>convertTextToOutlines</code>	boolean	如果为true，则所有文本都将转换为矢量路径；保留字体的外观。默认值： false
<code>flatteningBalance</code>	number (long)	扁平化Balance。范围：0.0到100.0。默认值： 100.0
<code>gradientResolution</code>	number (double)	梯度分辨率，以每英寸点数 (dpi) 为单位。范围：1.0至9600.0。默认值： 300.0
<code>overprint</code>	PDFOverprint	是保留，丢弃还是模拟叠印。 默认值： <code>PDFOverprint.PRESERVEPDFOVERPRINT</code>
<code>rasterizationResolution</code>	number (double)	光栅化分辨率，以每英寸点数 (dpi) 为单位。范围：1.0至9600.0。默认值： 300.0
<code>typename</code>	string	只读。对象的类名称。

设置打印展平

```
//创建一个新文档，向该文档中添加符号
//然后使用一系列展平的Balance设置进行打印

var docRef = documents.add();
var y = docRef.height - 30;
for(var i=0; i<(docRef.symbols.length); i++) {

    symbolRef = docRef.symbols[i];
    symbolItemRef1 = docRef.symbolItems.add(symbolRef);
    symbolItemRef1.top = y;
    symbolItemRef1.left = 100;
    y -= (symbolItemRef1.height + 10);
}
redraw();
// Create PrintFlattenerOptions object and assign to a PrintOptions object
var flatOpts = new PrintFlattenerOptions();
var printOpts = new PrintOptions();
printOpts.flattenerOptions = flatOpts;
// Set other print options
printOpts.ClipComplexRegions = true;
printOpts.GradientResoulution = 360;
printOpts.RasterizattonResotion = 360;

//以20的展平Balance增量打印当前文档
var i;
for(i=0; i<=100; i+=20) {
    flatOpts.flatteningBalance = i;
    activeDocument.print(printOpts);
}
```

PrintFontOptions

包含有关字体下载和用于打印文档的字体替换的信息。

PrintFontOptions properties

属性	值类型	注释
<code>downloadFonts</code>	PrintFontDownloadMode	字体下载模式。默认： <code>PrintFontDownloadMode.DOWNLOADSUBSET</code>
<code>fontSubstitution</code>	FontSubstitutionPolicy	字体替换策略。默认： <code>FontSubstitutionPolicy.SUBSTITUTEOBLIQUE</code>
<code>typename</code>	string	只读。对象的类名称。

使用字体选项打印

//创建一个新文档，添加文本，然后使用指定的字体选项进行打印。

```
var docRef = documents.add();
var pathRef = docRef.pathItems.rectangle(500,300,400,400);
var textRef = docRef.textFrames.areaText(pathRef);
textRef.contents = "Text example";
//创建PrintFontOptions对象并分配给PrintOptions对象
var fontOpts = new PrintFontOptions();
var printOpts = new PrintOptions();
printOpts.fontOptions = fontOpts;
//设置一些字体选项
fontOpts.downloadFonts = PrintFontDownloadMode.DOWNLOADNONE;
fontOpts.fontSubstitution = FontSubstitutionPolicy.SUBSTITUTEDEVICE;

//打印
activeDocument.print(printOpts);
```

PrintJobOptions

包含有关如何打印作业的信息。

PrintJobOptions properties

属性	值类型	注释
artboardRange	string	如果 printAllArtboards 为false, 则要打印的画板范围。默认值: 1-
bitmapResolution	number (double)	位图分辨率。最小值: 0.0。默认值: 0.0
collate	boolean	如果为true, 则整理打印页面。默认值: false
copies	number (long)	要打印的份数。最小值: 1。默认值: 1
designation	PrintArtworkDesignation	要打印的层/对象。默认: <code>PrintArtworkDesignation.VISIBL EPRINTABLELAYERS</code>
file	File	要打印到的文件。
name	string	打印作业名称。
printAllArtboards	boolean	指示是否打印所有画板。默认值: true
printArea	PrintingBounds	打印范围。 默认值: <code>PrintingBounds.ARTBOARDBOUNDS</code>
printAsBitmap	boolean	如果为true, 则打印为位图。默认值: false
reversePages	boolean	如果为true, 则以相反顺序打印页面。默认值: false
typename	string	只读。对象的类名称。

使用作业选项进行打印

```
//创建一个包含可见, 可打印,  
//不可见和不可打印项目的层的新文档, 然后以每个名称打印  
//查看使用不同作业选项的效果  
  
var docRef = documents.add();  
var textRef_0 = docRef.layers[0].textFrames.add();  
textRef_0.contents = "Visible and Printable";  
textRef_0.top = 600;  
textRef_0.left = 200;  
  
var layerRef_1 = docRef.layers.add();  
var textRef_1 = layerRef_1.textFrames.add();  
textRef_1.contents = "Visible and Non-Printable";  
textRef_1.top = 500;  
textRef_1.left = 250;  
layerRef_1.printable = false;  
  
var layerRef_2 = docRef.layers.add();  
var textRef_2 = layerRef_2.textFrames.add();  
textRef_2.contents = "Non-Visible";  
textRef_2.top = 400;  
textRef_2.left = 300;  
layerRef_2.visible = false;  
redraw();  
  
//使用各种作业选项进行打印  
var printJobOptions= new PrintJobOptions();  
var options = new PrintOptions();  
options.jobOptions = printJobOptions;  
  
printJobOptions.designation = PrintArtworkDesignation.ALLLAYERS;  
printJobOptions.reverse = true;  
docRef.print(options);  
  
printJobOptions.collate = false;  
printJobOptions.designation = PrintArtworkDesignation.VISIBLELAYERS;  
printJobOptions.reverse = false;  
docRef.print(options);  
  
printJobOptions.designation = PrintArtworkDesignation.VISIBLEPRINTABLELAYERS;  
var docPath = new File("~/printJobTest1.ps");  
printJobOptions.file = docPath;  
docRef.print(options);
```


PrintOptions

包含有关所有打印选项的信息，包括展平，颜色管理，坐标，字体和纸张。

PrintOptions properties

属性	值类型	注释
<code>colorManagementOptions</code>	PrintColorManagementOptions	打印颜色管理选项。
<code>colorSeparationOptions</code>	PrintColorSeparationOptions	打印分色选项。
<code>coordinateOptions</code>	PrintCoordinateOptions	打印坐标选项。
<code>flattenerOptions</code>	PrintFlattenerOptions	打印展平器选项。
<code>flattenerPreset</code>	string	透明度展平器预设名称。
<code>fontOptions</code>	PrintFontOptions	打印字体选项。
<code>jobOptions</code>	PrintJobOptions	打印作业选项。
<code>pageMarksOptions</code>	PrintPageMarksOptions	打印页面标记选项。
<code>paperOptions</code>	PrintPaperOptions	纸张选项。
<code>postScriptOptions</code>	PrintPostScriptOptions	打印PostScript选项。
<code>PPDName</code>	string	PPD名称。
<code>printerName</code>	string	打印机名称。
<code>printPreset</code>	string	打印样式。

设置打印选项

```
//创建一个新文档，添加符号，指定各种打印选项，
//将每个打印选项分配给一个PrintOptions对象，
//然后使用这些选项进行打印

//创建一个新文档并添加一些符号项
var docRef = documents.add();
var y = docRef.height - 30;
for(var i=0; i<(docRef.symbols.length); i++) {
    symbolRef = docRef.symbols[i];
    symbolItemRef1 = docRef.symbolItems.add(symbolRef);
    symbolItemRef1.top = y;
    symbolItemRef1.left = 100;
    y -= (symbolItemRef1.height + 10);
}
redraw();

//创建多个选项并分配给PrintOptions
var options = new PrintOptions();

var colorOptions = new PrintColorManagementOptions();
colorOptions.name = "ColorMatch RGB";
colorOptions.intent = PrintColorIntent.SATURATIONINTENT;
options.colorManagementOptions = colorOptions;

var printJobOptions= new PrintJobOptions();
printJobOptions.designation = PrintArtworkDesignation.ALLLAYERS;
printJobOptions.reverse = true;
options.jobOptions = printJobOptions;

var coordinateOptions = new PrintCoordinateOptions();
coordinateOptions.fitToPage = true;
options.coordinateOptions = coordinateOptions;

var flatOpts = new PrintFlattenerOptions();
flatOpts.ClipComplexRegions = true;
flatOpts.GradientResoulution = 60;
flatOpts.RasterizatonResotion = 60;
options.flattenerOptions = flatOpts;

//使用选项打印
docRef.print(options);
```

PrintPageMarksOptions

打印页面标记的选项。

PrintPageMarksOptions properties

属性	值类型	注释
bleedOffsetRect	array of 4 numbers	出血偏移矩形。
colorBars	boolean	如果为true, 请启用彩条打印。默认值: false
marksOffsetRect	array of 4 numbers	页面标记偏移矩形。
pageInfoMarks	boolean	如果为true, 则启用页面信息标记打印。默认值: false
pageMarksType	PageMarksTypes	页面标记样式。默认值: PageMarksType.Roman
registrationMarks	boolean	如果为true, 则应打印注册标记。默认值: false
trimMarks	boolean	如果为true, 则应打印修剪标记。默认值: false
trimMarksWeight	number (double)	描边的行程重量。最小值: 0.0。默认值: 0.125
typename	string	只读。对象的类名称。

设置页面标记打印选项

```
//创建一个PrintPageMarksOptions对象, 将其
//分配给一个PrintOptions对象, 然后打印当前文档。
```

```
var docRef = activeDocument;
var pageMarkOptions= new PrintPageMarksOptions();
var options = new PrintOptions();
options.pageMarksOptions = pageMarkOptions;

pageMarkOptions.colorBars = true;
pageMarkOptions.pageInfoMarks = true;
pageMarkOptions.registrationMarks = true;
pageMarkOptions.trimMarks = true;
docRef.print(options);
```

PrintPaperOptions

有关在打印作业中使用的纸张的信息。

PrintPaperOptions properties

属性	值类型	注释
height	number (double)	使用自定义纸张的自定义高度（以磅为单位）。默认值： 0.0
name	string	纸张的名称。
offset	number (double)	使用自定义纸张的自定义偏移量（以磅为单位）。默认值： 0.0
transverse	boolean	如果为true，请横切自定义纸张上的图稿（旋转90度）。默认值： false
typename	string	只读。对象的类名称。
width	number (double)	使用自定义纸张的自定义宽度（以磅为单位）。默认值： 0.0

设置打印纸选项

```
//创建一个新文档，添加一个路径项，应用图形样式
//然后使用指定的纸张选项进行打印

var docRef = documents.add();
var pathRef = docRef.pathItems.rectangle(600, 200, 200, 200);
docRef.graphicStyles[1].applyTo(pathRef);

var paperOpts = new PrintPaperOptions;
var printOpts = new PrintOptions;
printOpts.paperOptions = paperOpts;

var printerCount = printerList.length;
if (printerCount > 0) {
    //使用第一台打印机的第一张纸进行打印
    for (var i = 0; i < printerList.length; i++)
        if (printerList[i].printerInfo.paperSizes.length > 0)
            var printerRef = printerList[i];
    var paperRef = printerRef.printerInfo.paperSizes[0];
    if (printerRef.printerInfo.paperSizes.length > 0) {
        paperOpts.name = paperRef.name;
        printOpts.printerName = printerRef.name;

        docRef.print(printOpts);
    }
}
```

PrintPostScriptOptions

用于打印到PostScript打印机的选项。

PrintPostScriptOptions properties

属性	值类型	注释
binaryPrinting	boolean	如果为true, 则打印应为二进制模式。默认值: false
compatibleShading	boolean	如果为true, 请使用PostScript级别1兼容的渐变和渐变网格打印。默认值: false
forceContinuousTone	boolean	如果为true, 则强制连续音。默认值: false
imageCompression	PostScriptImageCompressionType	图像压缩类型。默认: PostScriptImageCompressionType.IMAGECOMPRESSIONNONE
negativePrinting	boolean	如果为true, 则以负片模式打印。默认值: false
postScriptLevel	PrinterPostScriptLevelEnum	PostScript语言级别。默认: PrinterPostScriptLevelEnum.LEVEL2
shadingResolution	number (double)	阴影分辨率。范围: 1.0至9600.0默认值: 300.0
typename	string	只读。对象的类名称。

设置PostScript打印选项

```
//打印具有各种后记级别的当前文档

//创建新的postscript选项对象, 分配给打印选项 var psOpts = new
PrintPostScriptOptions();
var printOpts = new PrintOptions();
printOpts.postScriptOptions = psOpts;
//分配PS级别, 打印
psOpts.postScriptLevel = PrinterPostScriptLevelEnum.PSLEVEL2;
activeDocument.print(printOpts);

psOpts.postScriptLevel = PrinterPostScriptLevelEnum.PSLEVEL3;
activeDocument.print(printOpts);
```

RasterEffectOptions

指定文档的光栅效果设置。所有属性都是可选的。

RasterEffectOptions properties

属性	值类型	注释
antiAliasing	boolean	如果为true，则应对图像进行抗锯齿处理。 默认值: false
clippingMask	boolean	如果为true，则为图像创建剪贴蒙版。默认值: false
colorModel	RasterizationColorModel	栅格化的颜色模型。默认: RasterizationColorModel.DEFAULTCOLORMODEL
convertSpotColors	boolean	如果为true，则所有专色都将转换为图像的原色。 默认值: false
padding	number (double)	在栅格化过程中要在对象周围添加的空白空间 (以磅为单位)。默认值: .0
resolution	number (double)	光栅化分辨率，以每英寸点数 (dpi) 为单 位。范围: 72.0至2400.0。默认值: 300.0
transparency	boolean	如果为true，则图像应使用透明度。默认值: false

RasterItem

文档中的位图艺术品。脚本可以从外部文件创建栅格项目，也可以通过使用重复方法复制现有栅格项目。

RasterItem properties

属性	值类型	注释
artworkKnockout	KnockoutState	此对象用于创建蒙版对象吗？如果是，则使用哪种蒙版类型。
bitsPerChannel	number (long)	只读。每个通道的位数。
blendingMode	BlendModes	合成对象时使用的混合模式。
boundingBox	array of 4 numbers	放置的艺术品的尺寸，与变换无关。
channels	number (long)	只读。通道数。
colorants	array of string	只读。光栅技术中使用的着色剂。
colorizedGrayscale	boolean	只读。如果为真，则光栅图为彩色灰度图像。
contentVariable	Variable	绑定到项目的content变量。
controlBounds	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
editable	boolean	只读。如果为true，则此项目是可编辑的。
embedded	boolean	如果为true，则光栅插图将嵌入插图中。
file	File	只读。包含插图的文件。
geometricBounds	array of 4 numbers	只读。项目的可见边界，包括笔触宽度。
height	number (double)	组项目的高度。
hidden	boolean	如果为true，则此项是隐藏的。
imageColorSpace	ImageColorSpace	只读。栅格图像的色彩空间。
isIsolated	boolean	如果为true，则隔离此对象。
layer	Layer	只读。此项所属的图层。
left	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。
locked	boolean	如果为true，则该项被锁定。
matrix	Matrix	放置的艺术品的转换矩阵。
name	string	此项的名称。

属性	值类型	注释
note	string	该项目的注释。
opacity	number (double)	对象的不透明度。范围：0.0至100.0
overprint	boolean	如果为真，则光栅图样会叠印。
parent	Layer or GroupItem	只读。此对象的父级。
position	array of 2 numbers	rasterItem 对象的左上角的位置（以磅为单位），格式为[x, y]。不包括行程重量。
selected	boolean	如果为true，则选择此项目。
sliced	boolean	如果为true，则将项目选中。默认值：false
status	RasterLinkState	链接图像的状态。
tags	Tags	只读。此项目中包含的标签。
top	number (double)	项目顶部的位置（以磅为单位，从页面底部开始测量）。
transparent	boolean	只读。如果为true，则光栅图是透明的。
typename	string	只读。被引用对象的类名。
uRL	string	分配给此项的Adobe URL标记的值。
visibilityVariable	Variable	绑定到项目的可见性变量。
visibleBounds	array of 4 numbers	只读。项目的可见边界，包括描边宽度。
width	number (double)	项目的宽度。
wrapInside	boolean	如果为true，则应将文本框架对象包装在该对象内。
wrapOffset	number (double)	在此对象上环绕文字时要使用的偏移量。
wrapped	boolean	如果为true，则将文本框架对象包装在该对象周围（文本框架必须在该对象上方）。
zOrderPosition	number	只读。该项目在包含该项目的组或层（父级）的堆叠顺序中的位置。

RasterItem methods

方法	参数类型	返回值	作用
colorize (rasterColor)	Color	Nothing	使用CMYK或RGB颜色为栅格项目着色。
duplicate ([relativeObject] [,insertionLocation])	object ElementPlacement	RasterItem	创建所选对象的副本。
move (relativeObject, insertionLocation)	object ElementPlacement	RasterItem	移动对象。
remove ()		Nothing	删除该对象。
resize (scaleX, scaleY [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,scaleAbout])	number (double) number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	缩放艺术品，其中scaleX是水平缩放因子，而scaleY是垂直缩放因子。100.0 = 100%。
rotate (angle [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,rotateAbout])	number (double) boolean boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转艺术品。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象。
trace ()		PluginItem	使用默认选项，将此对象的栅格图稿转换为矢量图稿。将栅格图稿重新排序为插件组的源图稿，并将其转换为一组类似于原始图像的填充和/或描边路径。创建并返回一个引用tracingObject对象的pluginItem对象。

方法	参数类型	返回值	作用
transform (transformationMatrix [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,transformAbout])	Matrix boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换艺术品。
translate ([deltaX] [,deltaY] [,transformObjects] [,transformFillPatterns] [,transformFillGradients] [,transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新放置艺术品，其中delX是水平偏移，而delY是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列艺术品的位置。

RasterItems

RasterItem对象的集合。

RasterItems properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

RasterItems methods

方法	参数类型	返回值	作用
getByName (name)	string	RasterItem	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	RasterItem	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

创建栅格项目

```
//从栅格文件在新文档中创建新的栅格项
// jpgFilePath包含jpg文件的完整路径和文件名
function createRasterItem(jpgFilePath) {
    var rasterFile = File(jpgFilePath);
    var myDoc = app.documents.add();
    var myPlacedItem = myDoc.placedItems.add();
    myPlacedItem.file = rasterFile;
    myPlacedItem.position = Array( 0, myDoc.height );
    myPlacedItem.embed();
}
```

查找和检查栅格项

//检查文档中第一个栅格项目的色彩空间,并在ESTK控制台中显示结果

```
if ( app.documents.length > 0 && app.activeDocument.rasterItems.length > 0 ) {  
    var rasterArt = app.activeDocument.rasterItems[0];  
  
    switch ( rasterArt.imageColorSpace ) {  
        case ImageColorSpace.CMYK:  
            $.writeln("The color space of the first raster item is CMYK");  
            break;  
  
        case ImageColorSpace.RGB:  
            $.writeln("The color space of the first raster item is RGB");  
            break;  
  
        case ImageColorSpace.GRAYSCALE:  
            $.writeln("The color space of the first raster item is GRAYSCALE");  
            break;  
    }  
}
```

RasterizeOptions

指定栅格化图稿时可能提供的选项。所有属性都是可选的。

RasterizeOptions properties

属性	值类型	注释
antiAliasingMethod	AntiAliasingMethod	抗锯齿方法的类型。默认： AntiAliasingMethod.ARTOPTIMIZED
backgroundBlack	boolean	如果为true，则在黑色背景（而不是白色）上进行光栅化。默认值： <code>false</code>
clippingMask	boolean	如果为true，则应为图像创建剪贴蒙版。默认值： <code>false</code>
colorModel	RasterizationColorModel	栅格化的颜色模型。默认： RasterizationColorModel.DEFAULTCOLOR MODEL
convertSpotColors	boolean	如果为true，则应将专色转换为图像的原色。默认值： <code>false</code>
convertTextToOutlines	boolean	如果为true，则在栅格化之前将所有文本转换为轮廓。默认值： <code>false</code>
includeLayers	boolean	如果为true，则生成的图像将包含图层属性（如不透明度和混合模式）。默认值： <code>false</code>
padding	number (double)	在栅格化过程中要在对象周围添加的空白空间（以磅为单位）。默认值： <code>.0</code>
resolution	number (double)	光栅化分辨率，以每英寸点数（dpi）为单位。范围： <code>72.0</code> 至 <code>2400.0</code> 。默认值： <code>300.0</code>
transparency	boolean	如果为true，则图像应使用透明度。默认值： <code>false</code>

RGBColor

RGB颜色规范，用于将RGB颜色应用于图层或艺术品。

如果文档的颜色空间是RGB，并且您使用CMYK为该文档中的页面项目指定了颜色值，则Illustrator会将CMYK颜色规格转换为RGB颜色规格。如果文档的颜色空间为CMYK，而您使用RGB指定颜色，则会发生相同的情况。由于这种翻译可能会丢失信息，因此您应该使用与文档的实际颜色空间匹配的类来指定颜色。

RGBColor properties

属性	值类型	注释
blue	number (double)	蓝色值。范围：0.0至255.0
green	number (double)	绿色值。范围：0.0至255.0
red	number (double)	红色值。范围：0.0至255.0
typename	string	只读。被引用对象的类名。

设置RGB颜色

//将当前文档中的默认填充颜色设置为黄色。

```
if ( app.documents.length > 0 ) {  
    // Define the new color  
    var newRGBColor = new RGBColor();  
  
    newRGBColor.red = 255;  
    newRGBColor.green = 255;  
    newRGBColor.blue = 0;  
    app.activeDocument.defaultFillColor = newRGBColor;  
}
```

Screen

将分色屏幕与要用于打印的信息相关联。

Screen properties

属性	值类型	注释
name	string	分色屏幕名称。
screenInfo	ScreenInfo	分色屏幕信息。
typename	string	只读。对象的类名称。

ScreenInfo

包含有关用于打印的分色屏的角度和频率的信息。

ScreenInfo properties

属性	值类型	注释
angle	number (double)	The screen's angle in degrees.
defaultScreen	boolean	If true, it is the default screen.
frequency	number (double)	The screen's frequency.
typename	string	只读。对象的类名称。

获取屏幕信息

//在新的文本框中显示每个屏幕列表项的名称，角度和频率

```
var sInfo = "";
var docRef = documents.add();
if(PPDFileList.length == 0){
    var sInfo = "\r\t\tEmpty PPDFileList"
}
else{
    var ppdRef = PPDFileList[0];
    var ppdInfoRef = ppdRef.PPDInfo;
    sInfo += "\r\t\tScreen Objects for 1st PPD File:\r";
    sInfo += "\t\t" + ppdRef.name;
    var iScreens = ppdInfoRef.screenList.length;
    if(iScreens > 0){
        for(var c=0; c<iScreens; c++) {
            var screenRef = ppdInfoRef.screenList[c];
            sInfo += "\r\t\t";
            sInfo += screenRef.name;

            var screenInfoRef = screenRef.screenInfo;
            sInfo += ", Angle = ";
            sInfo += screenInfoRef.angle;
            sInfo += ", Frequency = ";
            sInfo += screenInfoRef.frequency;
            sInfo += "\r";
        }
    }
    else{
        sInfo += "\r\t\tEmpty ScreenList";
    }
}
var textRef = docRef.textFrames.add();
textRef.textRange.characterAttributes.size = 12;
textRef.contents = sInfo;
textRef.top = 600;
textRef.left = 30;
redraw();
```


ScreenSpotFunction

包含有关分色屏幕专色功能的信息，包括其在PostScript语言代码中的定义。

ScreenSpotFunction properties

属性	值类型	注释
name	string	分色屏点功能名称。
spotFunction	string	用PostScript命令表示的spot函数。
typename	string	只读。对象的类名称。

查找屏幕点功能

```
//在新的文本框中显示第一个PPD文件的屏幕显示功能。
var docRef = documents.add();
if(PPDFileList.length == 0){
    var sInfo = "\r\t\tEmpty PPDFileList"
}
else{

    var ppdRef = PPDFileList[0];
    var ppdInfoRef = ppdRef.PPDInfo;
    var sInfo = "\r\t\tScreenSpotFunctions for 1st PPD File:\r";
    sInfo += "\t\t" + ppdRef.name + "\r";
    var iScreenSpots = ppdInfoRef.screenSpotFunctionList.length;
    if(iScreenSpots > 0 ){
        for(var n=0; n<iScreenSpots; n++) {
            var screenSpotRef = ppdInfoRef.screenSpotFunctionList[n];
            sInfo += "\t\t";
            sInfo += screenSpotRef.name;
            sInfo += ", spotFunction: ";
            sInfo += screenSpotRef.spotFunction;
            sInfo += "\r";
        }
    }
    else{
        sInfo += "\t\tEmpty ScreenSpotFunctionList";
    }
}
var textRef = docRef.textFrames.add();
textRef.textRange.characterAttributes.size = 12;
textRef.contents = sInfo;
textRef.top = 600;
textRef.left = 30;
redraw();
```

Spot

[SpotColor](#) 对象中包含的自定义颜色定义。

如果在创建竞标时未指定任何属性，那么将提供默认值。但是，如果指定颜色，则必须使用与文档相同的颜色空间CMYK或RGB。否则，将导致错误。新位置将添加到“色板”面板中的色板列表的末尾。

Spot properties

属性	值类型	注释
<code>color</code>	Color	该专色的颜色信息。
<code>colorType</code>	ColorModel	此自定义颜色的颜色模型。
<code>name</code>	string	专色的名称。
<code>parent</code>	Document	只读。包含此专色的文档。
<code>spotKind</code>	SpotColorKind	专色对象中包含的颜色种类。
<code>typename</code>	string	只读。被引用对象的类名。

Spot methods

方法	参数类型	返回值	作用
<code>getInternalColor</code> ()		Color components	获取斑点的内部颜色。
<code>remove</code> ()		Nothing	删除该对象。

创建新的专色

//在当前文档中创建一个新的专色，然后对该颜色应用80%的色彩

```
if ( app.documents.length > 0 ){  
    var doc = app.activeDocument;  
    // Create the new spot  
    var newSpot = doc.spots.add();  
    // Define the new color value  
    var newColor = new 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 = ColorModel.SPOT;  
    newSpot.color = newColor;  
    var newSpotColor = new SpotColor();  
    newSpotColor.spot = newSpot;  
    newSpotColor.tint = 80;  
}
```

SpotColor

颜色类，用于以指定的色度值应用专色的颜色值。可以在带有颜色对象的任何属性中使用。

SpotColor properties

属性	值类型	注释
spot	Spot	对定义颜色的专色对象的引用。
tint	number (double)	颜色的色彩。范围：0.0至100.0
typename	string	只读。被引用对象的类名。

Spots

文档中SpotColor对象的集合。

Spots properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

Spots methods

方法	参数类型	返回值	作用
add ()		Spot	创建一个新对象。
getByName (name)	string	Spot	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	Spot	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

去除专色

//从当前文档中删除所有专色

```
if ( app.documents.length > 0 ) {
var spotCount = app.activeDocument.spots.length;
  if (spotCount > 0) {
    app.activeDocument.spots.removeAll();
  }
}
```

创建和应用专色

//在当前文档中定义并应用新的专色，然后将该颜色应用到第一个路径项

```
if ( app.documents.length > 0 && app.activeDocument.pathItems.length > 0 ) {  
    //定义新的颜色值  
    newRGBColor = new RGBColor();  
    newRGBColor.red = 255;  
    newRGBColor.green = 0;  
    newRGBColor.blue = 0;  
  
    //创建新专色  
    var newSpot = app.activeDocument.spots.add();  
    //将新的SpotColor定义为RGB颜色的80%  
    newSpot.name = "Scripted Red spot";  
    newSpot.tint = 80;  
    newSpot.color = newRGBColor;  
  
    //将新专色的50%色彩应用于最前面的路径项。  
  
    //创建一个专色对象，设置色彩值  
    var newSpotColor = new SpotColor();  
    newSpotColor.spot = newSpot;  
    newSpotColor.tint = 50;  
    //使用专色设置填充颜色  
    var frontPath = app.activeDocument.pathItems[0];  
    frontPath.filled = true;  
    frontPath.fillColor = newSpotColor;  
}
```

Story

由文本范围指定的连续文本块。一个Story可以包含一个或多个文本框架。如果有多个，则将多个文本框架链接在一起以形成一个Story。

Story properties

属性	值类型	注释
<code>characters</code>	Characters	只读。这个Story中的所有角色。
<code>insertionPoints</code>	InsertionPoints	只读。这个Story中的所有插入点。
<code>length</code>	number (long)	只读。Story中的角色数量。
<code>lines</code>	Lines	只读。这个Story中的所有台词。
<code>paragraphs</code>	Paragraphs	只读。这个Story中的所有段落。
<code>parent</code>	object	只读。对象的容器。
<code>textFrames</code>	TextFrameItems	只读。这个Story中的文字框架项目。
<code>textRange</code>	TextRange	只读。Story的文字范围。
<code>textRanges</code>	TextRanges	只读。Story中的所有文字。
<code>textSelection</code>	array of TextRange	只读。选定的文本在Story中。
<code>typename</code>	string	只读。对象的类名称。
<code>words</code>	Words	只读。Story中的所有单词。

将文本框架插入Story

```
//创建一个故事，该故事流过2个文本框架，并创建另一个故事，

//在第三个文本框架中显示

//创建一个新文档并添加2个区域的TextFrames
var docRef = documents.add();
var itemRef1 = docRef.pathItems.rectangle(600, 200, 50, 30);
var textRef1 = docRef.textFrames.areaText(itemRef1);
textRef1.selected = true;

//创建第二个文本框架，并将其链接到第一个
var itemRef2 = docRef.pathItems.rectangle(550, 300, 50, 200);
var textRef2 = docRef.textFrames.areaText(itemRef2,
TextOrientation.HORIZONTAL, textRef1);
textRef2.selected = true;

//向第一个TextFrame添加足够的文本，

//以使其流向第二个TextFrame。
textRef1.contents = "This is two text frames linked together as one story";
redraw();

//创建第三个文本框并计算故事
var textRef3 = docRef.textFrames.add();
textRef3.contents = "Each unlinked textFrame adds a new story."
textRef3.top = 650;
textRef3.left = 200;
redraw();
```


Stories

文档中Story对象的集合。

Stories properties

属性	值类型	注释
length	number	只读。集合中元素的数量。
parent	object	只读。对象的容器。
typename	string	只读。对象的类名称。

Stories methods

方法	参数类型	返回值	作用
index (itemKey)	string, number	Story	从集合中获取一个元素。

Swatch

文档中包含的色样定义。这些色板与Illustrator用户界面中的色板调色板相对应。脚本可以创建一个新的色板。该色板可以保存所有类型的颜色数据，例如图案，渐变，CMYK，RGB，灰色和专色。

Swatch properties

属性	值类型	注释
color	Color	此色板的颜色信息。
name	string	色板的名称。
parent	Document	只读。包含此样本的文档。
typename	string	只读。被引用对象的类名。

Swatch methods

方法	参数类型	返回值	作用
remove ()		Nothing	删除该对象。

修改色板

//更改最后一个色板的名称

```
if ( app.documents.length > 0 && app.activeDocument.swatches.length > 0 ) {
    var lastIndex = app.activeDocument.swatches.length - 1;
    var lastSwatch = app.activeDocument.swatches[lastIndex];
    lastSwatch.name = "TheLastSwatch";
}
```

Swatches

文档中的Swatch对象的集合。

Swatches properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

Swatches methods

方法	参数类型	返回值	作用
add ()		Swatch	创建一个新的Swatch对象。
getByName (name)	string	Swatch	获取具有指定名称的集合中的一个元素。
getSelected ()		List of Swatch	获取文档中的选定色板。
index (itemKey)	string, number	Swatch	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

查找和删除样本

//从当前文档中删除样本4

```
if ( app.documents.length > 0 ) {

    if (app.activeDocument.swatches.length > 4)
    {
        swatchToDelete = app.activeDocument.swatches[3];
        swatchToDelete.remove();
    }
}
```

SwatchGroup

一组Swatch对象。

SwatchGroup properties

属性	值类型	注释
name	string	色板组的名称。
parent	object	只读。包含符号对象的对象。
typename	string	只读。被引用对象的类名。

SwatchGroup methods

方法	参数类型	返回值	作用
addSpot (spot)	Spot	Nothing	将现货色板添加到色板组。
addSwatch (swatch)	Swatch	Nothing	将色板添加到色板组。
getAllSwatches ()		List of Swatch	获取色板组中所有色板的列表。
remove ()		Nothing	删除该对象。
removeAll ()		Nothing	删除集合中的所有元素。

SwatchGroups

SwatchGroup对象的集合。

SwatchGroups properties

属性	值类型	注释
<code>length</code>	<code>number</code>	只读。集合中的对象数
<code>parent</code>	<code>object</code>	只读。此对象的父级。
<code>typename</code>	<code>string</code>	只读。被引用对象的类名。

SwatchGroups methods

方法	参数类型	返回值	作用
<code>add</code>		SwatchGroup	创建一个色板组。
<code>getByName</code> (name)	<code>string</code>	SwatchGroup	获取具有指定名称的集合中的第一个元素。
<code>removeAll</code> ()		Nothing	删除集合中的所有元素。

Symbol

一种艺术作品，该艺术作品存储在“符号”调板中，并且可以在文档中重复使用一次或多次，而无需复制艺术作品数据。符号包含在文档中。文档中Symbol的实例与SymbolItem对象相关联，该对象存储艺术品对象的属性。

Symbol properties

属性	值类型	注释
name	string	符号的名称。
parent	object	只读。包含符号对象的对象。
typename	string	只读。被引用对象的类名。

Symbol methods

方法	参数类型	返回值	作用
duplicate ()		Symbol	创建此对象的副本。
remove ()		Nothing	删除该对象。

SymbolItem

通过将其添加到“符号”调板使其可重用的艺术作品。**SymbolItem**链接到创建它的**Symbol**，如果您修改关联的**Symbol**对象，则**SymbolItem**会更改。

SymbolItem properties

属性	值类型	注释
artworkKnockout	KnockoutState	此对象用于创建蒙版对象吗？如果是，则使用哪种蒙版类型。
blendingMode	BlendModes	合成对象时使用的混合模式。
controlBounds	array of 4 numbers	只读。对象的边界，包括笔触宽度和控件。
editable	boolean	只读。如果为true，则此项目是可编辑的。
geometricBounds	array of 4 numbers	只读。除笔划宽度以外的对象边界。
height	number (double)	组项目的高度。
hidden	boolean	如果为true，则此项是隐藏的。
isIsolated	boolean	如果为true，则隔离此对象。
layer	Layer	只读。此项所属的图层。
left	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。
locked	boolean	If true, this item is locked.
name	string	此项的名称。
note	string	该项目的注释。
opacity	number (double)	对象的不透明度。范围：0.0至100.0
parent	Layer Or GroupItem	只读。此对象的父级。
position	array of 2 numbers	symbolItem 对象左上角的位置（以磅为单位），格式为[x, y]。不包括行程重量。
selected	boolean	如果为true，则选择此项目。
sliced	boolean	如果为true，则将项目选中。默认值：false
symbol	Symbol	用于创建此 symbolItem 的符号。
tags	Tags	只读。此项目中包含的标签。
top	number (double)	项目顶部的位置（以磅为单位，从页面底部开始测量）。

属性	值类型	注释
typename	string	只读。被引用对象的类名。
uRL	string	分配给此项的Adobe URL标记的值。
visibilityVariable	Variable	绑定到项目的可见性变量。
visibleBounds	array of 4 numbers	只读。项目的可见边界，包括笔触宽度。
width	number (double)	项目的宽度。
wrapInside	boolean	如果为true，则应将文本框架对象包装在该对象内。
wrapOffset	number (double)	在此对象上环绕文字时要使用的偏移量。
wrapped	boolean	如果为true，则将文本框架对象包装在该对象周围（文本框架必须在该对象上方）。
zOrderPosition	number	只读。该项目在包含该项目的组或层（父级）的堆叠顺序中的位置。

SymbolItem methods

方法	参数类型	返回值	作用
duplicate ([relativeObject] [, insertionLocation])	object ElementPlacement	SymbolItem	创建所选对象的副本。
move (relativeObject, insertionLocation)	object ElementPlacement	SymbolItem	移动对象。
remove ()		Nothing	删除该对象。
resize (scaleX, scaleY [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, scaleAbout])	number (double) number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	缩放艺术品，其中scaleX是水平缩放因子，而scaleY是垂直缩放因子。100.0 = 100%。

方法	参数类型	返回值	作用
rotate (angle [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, rotateAbout])	number (double) boolean boolean boolean Transformation	Nothing	相对于当前旋转旋转艺术品。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象
transform (transformationMatrix [, changePositions] [, changeFillPatterns] [, changeFillGradients] [, changeStrokePattern] [, changeLineWidths] [, transformAbout])	Matrix boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换艺术品。
translate ([deltaX] [, deltaY] [, transformObjects] [, transformFillPatterns] [, transformFillGradients] [, transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新放置艺术品，其中delX是水平偏移，而delY是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列艺术品的位置。

SymbolItems

文档中SymbolItem对象的集合。

SymbolItems properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

SymbolItems methods

方法	参数类型	返回值	作用
add (symbol)	Symbol	SymbolItem	创建指定符号的实例。
getByName (name)	string	SymbolItem	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	SymbolItem	从集合中获取一个元素。
removeAll ()		Nothing	删除集合中的所有元素。

创建符号项目

```
// Creates a new document then adds each of
// the documents symbols to the document

var docRef = documents.add();
var y = 750;
var x = 25;
var iCount = docRef.symbols.length;
for(var i=0; i<iCount; i++) {
    symbolRef = docRef.symbols[i];
    symbolItemRef1 = docRef.symbolItems.add(symbolRef);
    symbolItemRef1.top = y;
    symbolItemRef1.left = x;
    y-=(symbolItemRef1.height + 20);
    if( (y) <= 60 ) {
        y = 750;
        x+= 190;
    }
}
```

Symbols

文档中Symbol对象的集合。

Symbols properties

属性	值类型	注释
length	number	只读。集合中的对象数
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

Symbols methods

方法	参数类型	返回值	作用
add (sourceArt, [registrationPoint])	PageItem SymbolRegistrationPoint	Symbol	返回使用以下任何一种源艺术作品创建的符号对象： CompoundPathItems GroupItems MeshItems NonNativeItems PageItems PathItems RasterItems SymbolItems TextFrameItems 默认注册点是 SymbolCenterPoint.
index (itemKey)	string, number	Symbol	从集合中获取一个元素。
getByName (name)	string	Symbol	获取具有指定名称的集合中的第一个元素。
removeAll ()		Nothing	删除集合中的所有元素。

创建一个符号

```
//从每种图形样式创建一个路径项
//然后将每个项添加为新符号

var docRef = documents.add();
var y = 750;
var x =25;

var iCount = docRef.graphicStyles.length;
for(var i=0; i<iCount; i++) {
    var pathRef = docRef.pathItems.rectangle( y, x, 20,
20 ); docRef.graphicStyles[i].applyTo(pathRef);
    // are we at bottom?
    if( (y-=60) <= 60 ) {
        y = 750; // go back to the top.
        x+= 200
    }
    redraw();
    docRef.symbols.add(pathRef);
}
```

TabStopInfo

有关选项卡的对齐方式，位置和其他详细信息的信息在“段落属性”对象中停止。

TabStopInfo properties

属性	值类型	注释
alignment	TabStopAlignment	制表位的对齐方式。默认值：左
decimalCharacter	string	用于十进制制表符的字符停止。默认值：。
leader	string	前导点字符。
position	number (double)	制表位的位置以点表示。默认值：0.0
typename	string	只读。对象的类名称。

显示制表位停止信息

//显示在当前文档的每个文本框中找到的制表位停止信息，//如果有的话。

```
docRef = app.activeDocument;
var tabRef;
var sData = "Tab Stops Found \rTabStop Leader\t\tTabStop Position\r";
var textRef = docRef.textFrames;

for( var i=0 ; i < textRef.length; i++ ) {

    // Get all paragraphs in the textFrames
    paraRef = textRef[i].paragraphs;
    for ( p=0 ; p < paraRef.length ; p++ ) {
        // Get para attributes for all textRanges in paragraph
        attrRef = paraRef[p].paragraphAttributes;
        tabRef = attrRef.tabStops;
        if ( tabRef.length > 0 ) {
            for(var t=0; t<tabRef.length; t++){
                sData += "\t" + tabRef[t].leader + "\t\t";
                sData += "\t\t" + tabRef[t].position + "\r";
            } // end for
        } // end if
    } // end for
} // end for
var newTF = docRef.textFrames.add();
newTF.contents = sData;
newTF.top = 400;
newTF.left = 100;
redraw();
```

Tag

与特定艺术品相关的标签。标签使您可以为文档中的任何页面项目分配无限数量的键值对。

Tag properties

属性	值类型	注释
name	string	标签名称。
parent	object	只读。包含此标签的对象。
typename	string	只读。被引用对象的类名。
value	string	数据存储在该标签中。

Tag methods

方法	参数类型	返回值	作用
remove ()		Nothing	删除该对象。

使用标签

//查找与所选艺术品相关的标签，//在单独的文档中显示名称和值

```

if ( app.documents.length > 0 ) {

    doc = app.activeDocument;
    if ( doc.selection.length > 0 ) {
        for ( i = 0; i < selection.length; i++ ) {
            selectedArt = selection[0];
            tagList = selectedArt.tags;
            if (tagList.length == 0) {
                var tempTag = tagList.add();
                tempTag.name = "OneWord";
                tempTag.value = "anything you want";
            }
            // Create a document and add a line of text per tag
            reportDocument = app.documents.add();
            top_offset = 400;
            for ( i = 0; i < tagList.length; i++ ) {
                tagText = tagList[i].value;
                newItem = reportDocument.textFrames.add();
                newItem.contents = "Tag: (" + tagList[i].name +
                    " , " + tagText + ")";
                newItem.position = Array(100, top_offset);
                newItem.textRange.size = 24;
                top_offset = top_offset - 20;
            }
        }
    }
}

```

```
}  
}
```

Tags

Tag对象的集合。

Tags properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

Tags methods

方法	参数类型	返回值	作用
add ()		Tag	创建一个新的Tag对象。
getByName (name)	string	Tag	获取具有指定名称的集合中的第一个元素。
index (itemKey)	string, number	Tag	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

设置标签值

//将标签添加到当前文档中的所有RasterItems和PlacedItems

```

if ( app.documents.length > 0 ) {
    doc = app.activeDocument;
    if ( doc.placedItems.length + doc.rasterItems.length > 0 ) {
        for ( i = 0; i < doc.pageItems.length; i++ ) {
            imageArt = doc.pageItems[i];
            if ( imageArt.typename == "PlacedItem"
                || imageArt.typename == "RasterItem" ) {
                // Create a new Tag with the name AdobeURL and the
                // value of the www link
                urlTAG = imageArt.tags.add();
                urlTAG.name = "AdobeWebSite";
                urlTAG.value = "http://www.adobe.com/";
            }
        }
    }
    else {
        alert( "No placed or raster items in the document" );
    }
}

```


TextFont

有关文档中字体的信息，可在CharacterAttributes对象中找到。

TextFont properties

属性	值类型	注释
family	string	只读。字体的族。
name	string	只读。字体的全名。
parent	object	只读。对象的容器。
style	string	只读。字体的样式名称。
typename	string	只读。对象的类名称。

设置文字字体

//将文档中所有文本的字体设置为第一个字体

```
if ( app.documents.length > 0 ) {  
    // Iterate through all text art and apply font 0  
    for ( i = 0; i < app.activeDocument.textFrames.length; i++ ) {  
        textArtRange = app.activeDocument.textFrames[i].textRange;  
        textArtRange.characterAttributes.textFont = app.textFonts[0];  
    }  
}
```

TextFonts

TextFont对象的集合。

TextFonts properties

属性	值类型	注释
length	number	只读。集合中元素的数量。
parent	object	只读。对象的容器。
typename	string	只读。对象的类名称。

TextFonts methods

方法	参数类型	返回值	作用
index (itemKey)	string, number	TextFont	从集合中获取元素。
getByName (name)	string	TextFont	使用提供的名称获取集合中的一个元素。

查找字体

//创建一个新的A3尺寸的文档，并显示可用字体的列表，直到文档填满。

```
var edgeSpacing = 10;
var columnSpacing = 230;
var docPreset = new DocumentPreset;
docPreset.width = 1191.0;
docPreset.height = 842.0

var docRef = documents.addDocument(DocumentColorSpace.CMYK, docPreset);
var sFontNames = "";
var x = edgeSpacing;
var y = (docRef.height - edgeSpacing);

var iCount = textFonts.length;
for(var i=0; i<iCount; i++) {

    sFontName = textFonts[i].name;
    sFontName += " ";
    sFontNames = sFontName + textFonts[i].style;

    var textRef = docRef.textFrames.add();
    textRef.textRange.characterAttributes.size = 10;
    textRef.contents = sFontNames;
    textRef.top = y;
    textRef.left = x;

    //检查文本框是否会偏离文档的边缘
```

```
    if ((x + textRef.width) > docRef.width) {
        textRef.remove();
        iCount = i;
        break;
    }
    else {
        //显示文字框
        textRef.textRange.characterAttributes.textFont =
textFonts.getByname(textFonts[i].name);
        redraw();

        if( (y--=(textRef.height)) <= 20 ) {
            y = (docRef.height - edgeSpacing);
            x += columnSpacing;
        }
    }
}
```

TextFrameItem

用于显示文本的基本美术作品。在用户界面上，这是使用“文本”工具创建的文本。Illustrator中有三种文字艺术：点文字，路径文字和区域文字。类型由文本框的 [kind](#) 属性指示。

创建文本框架时，还将创建一个 [Story](#) 对象。但是，线程文本框架将框架组合成单个故事对象。要链接框架，请使用 [nextFrame](#) 或 [previousFrame](#) 属性。

TextFrameItem properties

属性	值类型	注释
anchor	array of 2 numbers	锚点的位置，点文本的基线的起点。
antialias	TextAntialias	文本中使用的抗锯齿类型。
characters	Characters	只读。此文本框中的所有字符。
columnCount	number (long)	文本框中的列数（仅区域文本）。
columnGutter	number (double)	文本框中的列装订线（仅区域文本）。
contents	string	文本字符串。
contentVariable	Variable	绑定到此文本框架项目的content变量。
endTValue	number (double)	文本在路径上的结束位置，是相对于路径段的值（仅路径文本）。
flowLinksHorizontally	boolean	如果为true，请首先在链接的框架之间水平移动文本（仅区域文本）。
insertionPoints	InsertionPoints	只读。此文本范围内的所有插入点。
kind	TextType	只读。文本框项目的类型（区域，路径或点）。
lines	Lines	只读。此文本框中的所有行。
matrix	Matrix	只读。此文本框的转换矩阵。
nextFrame	TextFrameItem	此后的链接文本框架。
opticalAlignment	boolean	如果为true，则光学对齐功能处于活动状态。
orientation	TextOrientation	文本的方向。
paragraphs	Paragraphs	只读。此文本框中的所有段落。
parent	Layer Or GroupItem	只读。此对象的父级。
previousFrame	TextFrameItem	The linked text frame preceding this one.
rowCount	number (long)	The row count in the text frame (area text only).

属性	值类型	注释
rowGutter	number (double)	文本框中的行装订线（仅区域文本）。
spacing	number (double)	间距量。
startTValue	number (double)	文本沿路径的开始位置，相对于路径段的值（仅路径文本）。
story	Story	只读。文本框架所属的故事。
textPath	TextPath	与文本框架关联的路径项。注意：仅当 kind 为区域或路径时有效。
textRange	TextRange	只读。文本框架的文本范围。
textRanges	TextRanges	只读。此文本框中的所有文本。
textSelection	array of TextRange	只读。文本框中的选定文本范围。
typename	string	只读。被引用对象的类名。
words	Words	只读。此文本框中的所有单词。

TextFrameltem methods

方法	参数类型	返回值	作用
convertAreaObjectToPointObject ()		TextFrameItem	将区域类型的文本框架转换为点型文本框架。
convertPointObjectToAreaObject ()		TextFrameItem	将点型文本框转换为区域型文本框。
createOutline ()		GroupItem	将文本框架中的文本转换为轮廓。
duplicate ([relativeObject] [, insertionLocation])	object ElementPlacement	TextRange	创建所选对象的副本。
move (relativeObject, insertionLocation)	object ElementPlacement	TextRange	移动对象。
remove ()		Nothing	删除该对象。

方法	参数类型	返回值	作用
resize (scaleX, scaleY [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,scaleAbout])	number (double) number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	缩放艺术品，其中 scaleX 是水平缩放因子，而 scaleY 是垂直缩放因子。100.0 = 100%。
rotate (angle [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,rotateAbout])	number (double) boolean boolean boolean boolean number (double) Transformation	Nothing	相对于当前旋转旋转艺术品。如果角度值为正，则逆时针旋转对象；如果角度为负，则逆时针旋转对象。
transform (transformationMatrix [,changePositions] [,changeFillPatterns] [,changeFillGradients] [,changeStrokePattern] [,changeLineWidths] [,transformAbout])	Matrix boolean boolean boolean boolean number (double) Transformation	Nothing	通过应用变换矩阵来变换艺术品。
translate ([deltaX] [,deltaY] [,transformObjects] [,transformFillPatterns] [,transformFillGradients] [,transformStrokePatterns])	number (double) number (double) boolean boolean boolean boolean	Nothing	相对于当前位置重新放置艺术品，其中 delX 是水平偏移，而 delY 是垂直偏移。
zOrder (zOrderCmd)	ZOrderMethod	Nothing	按照此对象的组或图层（父级）的堆叠顺序排列艺术品的位置。

旋转文字艺术项目

```
//复制并旋转选定的文字艺术项目5次

if ( app.documents.length > 0 ) {
    selectedItems = app.activeDocument.selection;
    // make sure something is selected.
    if ( selectedItems.length > 0 ) {
        // The selection must be a text art item
        if ( selectedItems[0].typename == "TextFrame" ) {
            // Get the parent of the text art so new text art items
            // can be inserted in the same group or layer
            dupSrc = selectedItems[0];
            textContainer = dupSrc.parent;
            // Create 5 new versions of the text art each rotated a bit
            for ( i = 1; i <= 5; i++ ) {
                dupText = dupSrc.duplicate( textContainer,
                    ElementPlacement.PLACEATEND );
                dupText.rotate(180 * i/6);
            }
        }
    }
}
```

TextFrameItems

A collection of [TextFrameItem](#) objects.

TextFrameItems properties

属性	值类型	注释
length	number	只读。集合中元素的数量。
parent	object	只读。对象的容器。
typename	string	只读。对象的类名称。

TextFrameItems methods

方法	参数类型	返回值	作用
add ()		TextFrameItem	创建一个点文本框架项。
areaText (textPath [,orientation] [,baseFrame] [,postFix])	PathItem TextOrientation TextFrameItem boolean	TextFrameItem	创建一个区域文本框架项。
getByName (name)	string	TextFrameItem	使用提供的名称获取集合中的第一个元素。
index (itemKey)	string, number	TextFrameItem	从集合中获取一个元素。
pathText (textPath [,startTValue] [,endTValue] [,orientation] [,baseFrame] [,postFix])	PathItem number (double) number (double) TextOrientation TextFrameItem boolean	TextFrameItem	创建路径上的文本框架项目。
pointText (anchor [,orientation])	array of 2 numbers TextOrientation	TextFrameItem	创建一个点文本框项目
removeAll ()		Nothing	删除对象中的所有元素。

创建和修改文本框架

```
//创建一个带有显示路径, 区域和点文本的文本框架的文档,

//更改每个框架的内容, 然后删除第二个框架

//创建一个新文档
var docRef = documents.add();

//创建3个新的textFrames (区域, 线, 点)
//区域文本
var rectRef = docRef.pathItems.rectangle(700, 50, 100, 100);
var areaTextRef = docRef.textFrames.areaText(rectRef);
areaTextRef.contents = "TextFrame #1";
areaTextRef.selected = true;

//行文本
var lineRef = docRef.pathItems.add();
lineRef.setEntirePath( Array(Array(200, 700), Array(300, 550) ) );
var pathTextRef = docRef.textFrames.pathText(lineRef);
pathTextRef.contents = "TextFrame #2";
pathTextRef.selected = true;

//点文字
var pointTextRef = docRef.textFrames.add();
pointTextRef.contents = "TextFrame #3";
pointTextRef.top = 700;
pointTextRef.left = 400;
pointTextRef.selected = true;
redraw();

//计算TextFrames
var iCount = docRef.textFrames.length;
var sText = "There are " + iCount + " TextFrames.\r"
sText += "Changing contents of each TextFrame.";

//更改每个内容
docRef.textFrames[0].contents = "Area TextFrame.";
docRef.textFrames[1].contents = "Path TextFrame.";
docRef.textFrames[2].contents = "Point TextFrame.";
redraw();

docRef.textFrames[1].remove();
redraw();

//再数一次
var iCount = docRef.textFrames.length;
```

TextPath

区域或路径文本的路径或路径列表。路由定义其几何形状的路径点组成。

TextPath properties

属性	值类型	注释
area	number (double)	只读。该路径的面积（以平方点为单位）。如果该区域为负，则该路径将逆时针缠绕。自相交路径可以包含相互抵消的子区域，即使该路径具有明显的面积，该子区域也使该值变为零。
blendingMode	BlendModes	合成对象时使用的混合模式。
clipping	boolean	如果为true，则应将此路径用作剪切路径。
closed	boolean	如果为true，则关闭此路径。
editable	boolean	只读。如果为true，则此项目是可编辑的。
evenodd	boolean	如果为true，则应使用奇数规则确定内部性。
fillColor	Color	路径的填充颜色。
filled	boolean	如果为true，则填充路径。
fillOverprint	boolean	如果为true，则应套印填充对象下的艺术品。
guides	boolean	如果为true，则此路径为参考线。
height	number (double)	组项目的高度。
left	number (double)	项目左侧的位置（以磅为单位，从页面左侧开始测量）。
note	string	分配给路径的注释文本。
opacity	number (double)	对象的不透明度。范围：0.0至100.0
parent	Layer OR GroupItem	只读。此对象的父级。
pathPoints	PathPoints	只读。此路径项中包含的路径点。
polarity	PolarityValues	路径的极性。
position	array of 2 numbers	textPathItem 对象左上角的位置（以磅为单位），格式为[x, y]。不包括行程重量。
resolution	number (double)	路径的分辨率，以每英寸点数（dpi）为单位。
selectedPathPoints	PathPoints	只读。路径中所有选定的路径点。

属性	值类型	注释
strokeCap	StrokeCap	线路封顶的类型。
strokeColor	Color	路径的笔触颜色。
stroked	boolean	如果为true, 则应绘制路径。
strokeDashes	object	短划线长度。设置为空对象 {}, 以获得实线。
strokeDashOffset	number (double)	距虚线图案的默认距离, 应从该距离开始。
strokeJoin	StrokeJoin	路径的关节类型。
strokeMiterLimit	number (double)	当默认笔触连接设置为“斜接”时, 此属性指定默认情况下何时将连接转换为斜角 (平方)。默认的斜接限制为4表示当点的长度达到笔划粗细的四倍时, 连接将从斜接连接切换为斜角连接。值1指定一个斜角连接。范围: 1到500。默认值: 4
strokeOverprint	boolean	如果为true, 则应套印描边对象下方的艺术品。
strokeWidth	number (double)	笔划的宽度。
top	number (double)	项目顶部的位置 (以磅为单位, 从页面底部开始测量)。
typename	string	只读。被引用对象的类名。
width	number (double)	项目的宽度。

TextPath methods

方法	参数类型	返回值	作用
setEntirePath (pathPoints)	array of [x, y] coordinate pairs	Nothing	使用指定为[x, y]坐标对的点的数组设置路径。

TextRange

特定文字艺术项目中的文字范围。`TextRange`使您可以访问包含在文字艺术项目中的文本。

TextRange properties

属性	值类型	注释
<code>characterAttributes</code>	CharacterAttributes	只读。文本范围的字符属性。
<code>characterOffset</code>	number (long)	第一个字符的偏移量。
<code>characters</code>	Characters	只读。此文本范围内的所有字符。
<code>characterStyles</code>	CharacterStyles	只读。文本范围内所有引用的字符样式。
<code>contents</code>	string	文本字符串。
<code>end</code>	Int32	文本范围的结束索引。
<code>insertionPoints</code>	InsertionPoints	只读。此文本范围内的所有插入点。
<code>kerning</code>	number (long)	控制两个字符之间的间距，以em的千分之一为单位。一个整数。
<code>length</code>	number (long)	长度（以字符为单位）。最小：0
<code>lines</code>	Lines	只读。此文本范围内的所有行。
<code>paragraphAttributes</code>	ParagraphAttributes	只读。文本范围的段落属性。
<code>paragraphs</code>	Paragraphs	只读。此文本中的所有段落。
<code>paragraphStyles</code>	ParagraphStyles	只读。文本范围内所有引用的段落样式。
<code>parent</code>	TextRange	只读。对象的容器。
<code>start</code>	Int32	文本范围的起始索引。
<code>story</code>	Story	只读。文字范围所属的故事。
<code>textRanges</code>	TextRanges	只读。此文本范围内的所有文本。
<code>textSelection</code>	array of TextRange	只读。选定的文本范围在文本范围内。
<code>typename</code>	string	只读。对象的类名。
<code>words</code>	Words	只读。该文本范围内包含的所有单词。

TextRange methods

方法	参数类型	返回值	作用
changeCaseTo (type)	CaseChangeType	Nothing	更改文本的大写。
deSelect ()		Nothing	取消选择文本范围。
duplicate ([relativeObject] [, insertionLocation])	object ElementPlacement	TextRange	创建此对象的副本。
getLocalCharOverridesJSON ()		String	获取字符替代的json表示形式。
getLocalParaOverridesJSON ()		String	获取段落替代的json表示形式。
getParagraphLength ()		Int32	获取文本范围的第一段的长度。
getTextRunLength ()		Int32	获取文本范围的第一个文本行的长度。
move (relativeObject, insertionLocation)	object ElementPlacement	TextRange	移动对象。
remove ()		Nothing	删除对象。
select ([addToDocument])	boolean	Nothing	选择文本范围。如果 addToDocument 为 true ，则将其添加到当前选择中；否则为 false 。否则替换当前选择。

处理文字

//通过更改每个字符的大小属性来更改当前文档中每个单词的第一个字符的大小

```

if ( app.documents.length > 0 ) {
  for ( i = 0; i < app.activeDocument.textFrames.length; i++ ) {
    text = app.activeDocument.textFrames[i].textRange;
    for ( j = 0 ; j < text.words.length; j++ ) {
      //each word is a textRange object
      textWord = text.words[j];
      // Characters are textRanges too.
      // Get the first character of each word and increase it's size.
      firstChars = textWord.characters[0];
      firstChars.size = firstChars.size * 1.5;
    }
  }
}

```

TextRanges

TextRange对象的集合。

TextRanges properties

属性	值类型	注释
length	number	只读。集合中元素的数量。
parent	object	只读。对象的容器。
typename	string	只读。对象的类名称。

TextRanges methods

方法	参数类型	返回值	作用
index (itemKey)	string, number	TextRange	从集合中获取元素
removeAll ()		Nothing	删除对象中的所有元素。

TracingObject

跟踪对象，它将源栅格图稿与通过跟踪创建的矢量图稿插件组相关联。脚本可以使用 `PlacedItem.trace` 或 `RasterItem.trace` 启动跟踪。

生成的 `PluginItem` 对象表示矢量图稿作品组，并且在跟踪属性中具有此对象。脚本可以通过调用应用程序的 `redraw` 方法来强制执行跟踪操作。

该操作是异步的，因此脚本应在创建跟踪对象之后但在访问其属性或扩展跟踪以将其转换为图稿项目组之前调用重绘。仅在第一个跟踪操作完成后，描述跟踪结果的只读属性才具有有效值。值为 0 表示该操作尚未完成。

TracingObject properties

属性	值类型	注释
<code>anchorCount</code>	number (long)	只读。跟踪结果中的锚点数。
<code>areaCount</code>	number (long)	只读。跟踪结果中的区域数。
<code>imageResolution</code>	number (real)	只读。源图像的分辨率，以每英寸像素为单位。
<code>parent</code>	object	只读。对象的容器。
<code>pathCount</code>	number (long)	只读。跟踪结果中的路径数。
<code>sourceArt</code>	PlacedItem Or RasterItem	用于创建关联的矢量图稿插件组的栅格图稿。
<code>tracingOptions</code>	TracingOptions	用于将光栅图稿转换为矢量图稿的选项。
<code>typename</code>	string	只读。对象的类名称。
<code>usedColorCount</code>	number (long)	只读。跟踪结果中使用的颜色数。

TracingObject methods

方法	参数类型	返回值	作用
expandTracing ([viewed])	boolean	GroupItem	将矢量艺术转换为新的组项目。新的 GroupItem 对象将替换文档中的 PluginItem 对象。默认情况下， viewed 为 false ，并且新组仅包含跟踪结果（填充或描边的路径）。如果已查看为真，则新组将保留为查看模式指定的其他信息，例如轮廓和覆盖图。删除此对象及其关联的 PluginItem 对象。应用于插件项目的所有组级别属性都将应用于新组项目的顶层。
releaseTracing ()		PlacedItem or RasterItem	将文档中的图稿还原为原始源栅格图稿，并删除跟踪的矢量图稿。返回用于创建跟踪的原始对象，并删除该对象及其关联的 PluginItem 对象。

TracingOptions

通过跟踪将栅格图稿转换为矢量图稿时使用的一组选项。

TracingOptions properties

属性	值类型	注释
<code>cornerAngle</code>	number (double)	原始图像中以转度为单位的清晰度，该清晰度被视为跟踪结果路径中的一个角。范围：0至180
<code>fills</code>	boolean	如果为true，则填充填充。填充或笔触中的至少一个必须为真。
<code>ignoreWhite</code>	boolean	如果为true，则忽略白色填充颜色。
<code>livePaintOutput</code>	boolean	如果为true，则结果为Live Paint艺术。虚假的是古典艺术。 注意：脚本仅应在准备后续扩展操作时设置此值。当此属性为true时，在画板上留下跟踪可能会导致应用程序意外行为。
<code>maxColors</code>	number (long)	自动调色板生成允许的最大颜色数。仅当tracingMode为彩色或灰度时使用。范围：2至256
<code>maxStrokeWeight</code>	number (double)	笔划为真时的最大笔划重量。范围：0.01至100.0
<code>minArea</code>	number (long)	跟踪的最小特征（以正方形像素为单位）。例如，如果为4，则将跟踪2像素宽乘2像素高的特征。
<code>minStrokeLength</code>	number (double)	笔触为true时，可以笔触的原始图像中的特征的最小长度（以像素为单位）。省略了较小的功能。范围：0.0到200.0。默认值：20.0
<code>outputToSwatches</code>	boolean	如果为true，则为跟踪结果创建的每种新颜色生成命名的颜色（色板）。仅当tracingMode为彩色或灰度时使用。
<code>palette</code>	string	用于跟踪的调色板的名称。如果字符串为空，请使用自动调色板。仅当tracingMode为彩色或灰度时使用。
<code>parent</code>	object	只读。对象的容器。
<code>pathFitting</code>	number (double)	跟踪的形状和原始像素形状之间的距离。较低的值会产生更紧密的路径拟合。较高的值将创建较宽松的路径拟合。范围：0.0至10.0

属性	值类型	注释
<code>preprocessBlur</code>	number (double)	预处理期间使用的模糊量，以像素为单位。模糊有助于减少跟踪结果中的小瑕疵和平滑锯齿状的边缘。范围：0.0至2.0
<code>preset</code>	string	只读。包含这些选项的预设文件的名称。
<code>resample</code>	boolean	如果为true，则在跟踪时重新采样。（此设置不会捕获到预设文件中。）放置或链接光栅图稿时，始终为true。
<code>resampleResolution</code>	number (double)	重新采样时使用的分辨率，以每英寸像素 (ppi) 为单位。较低的分辨率会提高跟踪操作的速度。（此设置未捕获到预设文件中。）
<code>strokes</code>	boolean	如果为true，则描边描边。填充或笔触中的至少一个必须为真。仅在tracingMode为黑白时使用。
<code>threshold</code>	number (long)	黑白跟踪的阈值。灰度值大于此值的所有像素都将转换为黑色。仅在tracingMode为黑白时使用。范围：0至255
<code>tracingMode</code>	TracingModeType	用于跟踪的颜色模式。
<code>typename</code>	string	只读。对象的类名称。
<code>viewRaster</code>	ViewRasterType	光栅图像预览的视图。（此设置未捕获到预设文件中。）
<code>viewVector</code>	ViewVectorType	向量结果预览的视图。（此设置未捕获到预设文件中。）

TracingOptions methods

方法	参数类型	返回值	作用
<code>loadFromPreset</code> (presetName)	string	boolean	从Application.tracingPresetList数组中找到的指定预设中加载一组选项。
<code>storeToPreset</code> (presetName)	string	boolean	将这组选项保存在指定的预设中。使用在Application.tracingPresetList数组中找到的名称或新名称来创建新的预设。对于现有预设，将覆盖未锁定的预设并返回true。如果预设已锁定，则返回false。

Variable

可以导入或导出的文档级变量。

变量是用于创建数据驱动图形的动态对象。有关示例，请参见数据集。在Illustrator中通过“变量”面板访问变量。

Variable properties

属性	值类型	注释
kind	VariableKind	变量的类型。
name	string	变量的名称。
pageItems	PageItems	只读。变量中的所有插图。
parent	object	只读。包含变量的对象。
typename	string	只读。被引用对象的类名。

Variable methods

方法	参数类型	返回值	作用
remove ()		Nothing	从变量集合中删除变量。

Variables

文档中变量对象的集合。有关如何创建变量的示例，请参见 [Using variables and datasets](#)。使用变量和数据集。

Variables properties

属性	值类型	注释
length	number	只读。文档中的变量数
parent	object	只读。包含变量集合的对象。
typename	string	只读。被引用对象的类名。

Variables methods

方法	参数类型	返回值	作用
add ()		Variable	将新变量添加到集合中。
getByName (name)	string	Variable	使用提供的名称获取集合中的第一个元素。
index (itemKey)	string, number	Variable	从集合中获取元素。
removeAll ()		Nothing	删除集合中的所有元素。

View

Illustrator文档中的文档视图，它表示文档上的窗口视图。脚本无法创建新视图，但是可以修改现有视图的某些属性，包括中心点，屏幕模式和缩放。

View properties

属性	值类型	注释
bounds	array of 4 numbers	只读。相对于当前文档边界的此视图的边界矩形。
centerPoint	array of 2 numbers	此视图相对于当前文档边界的中心点。
parent	Document	只读。包含此视图的文档。
screenMode	ScreenMode	此视图的显示模式。
typename	string	只读。被引用对象的类名。
zoom	number (double)	此视图的缩放系数，其中100.0是100%。

将视图设置为全屏

```
//将当前文档的屏幕模式设置为全屏  
  
if ( app.documents.length > 0 ) {  
    app.documents[0].views[0].screenMode = ScreenMode.FULLSCREEN;  
}
```

Views

文档中View对象的集合。

Views properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

Views methods

方法	参数类型	返回值	作用
index (itemKey)	string, number	View	从集合中获取一个元素。

Words

文本项中单词的集合，其中每个单词都是一个TextRange对象。元素未命名；您必须按索引访问它们。

Words properties

属性	值类型	注释
length	number	只读。集合中的对象数。
parent	object	只读。此对象的父级。
typename	string	只读。被引用对象的类名。

Words methods

方法	参数类型	返回值	作用
add (contents [, relativeObject] [, insertionLocation])	string TextFrameItem ElementPlacement	TextRange	将单词添加到当前文档中的指定位置。如果未指定位置，则将其添加到当前单词选择或插入点之后的包含文本框架。
addBefore (contents)	string	TextRange	在当前单词选择或插入点之前添加单词。
index (itemKey)	number	TextRange	从集合中获取一个元素。
removeAll ()		Nothing	删除此集合中的所有元素。

数单词

//计算当前文档中的所有单词并将总数存储在numWords中

```
if ( app.documents.length > 0 ) {
  numWords = 0;
  for ( i = 0; i < app.activeDocument.textFrames.length; i++) {
    numWords += app.activeDocument.textFrames[i].words.length;
  }
}
```

将属性应用于单词

//创建一种新的洋红色，并将该颜色应用于符合特定条件的所有单词

```
if ( app.documents.length > 0 && app.activeDocument.textFrames.length > 0 ) {  
  
    // Create the color to apply to the words  
    wordColor = new RGBColor();  
    wordColor.red = 255;  
    wordColor.green = 0;  
    wordColor.blue = 255;  
    //设置要查找的单词的值  
    searchWord1 = "the";  
    searchWord2 = "The";  
    searchWord3 = "THE";  
    //遍历文档中的所有单词  
    //并为与searchWord匹配的单词上色  
    for ( i = 0; i < app.activeDocument.textFrames.length; i++ ) {  
        textArt = activeDocument.textFrames[i];  
        for ( j = 0; j < textArt.words.length; j++ ) {  
            word = textArt.words[j];  
            if ( word.contents == searchWord1 || word.contents == searchWord2 ||  
                word.contents == searchWord3 ) {  
                word.filled = true;  
                word.fillColor = wordColor;  
            }  
        }  
    }  
}
```


脚本常量

本章列出并描述了为与Illustrator JavaScript属性和方法一起使用而定义的枚举。

常量	值	注释
AlternateGlyphsForm		
	DEFAULTFORM	THIRDWIDTH
	TRADITIONAL	QUARTERWIDTH
	EXPERT	FULLWIDTH
	JIS78FORM	PROPORTIONALWIDTH
	JIS83FORM	JIS90FORM
	HALFWIDTH	JIS04FORM
AntiAliasingMethod		
	None	TYPEOPTIMIZED
	ARTOPTIMIZED	
		栅格化中使用的抗锯齿方法的类型。
		▶ None — 禁止抗锯齿。
		▶ ARTOPTIMIZED — 针对图稿进行优化。
		▶ TYPEOPTIMIZED — 针对类型对象进行优化。
ArtClippingOption		
	OUTPUTARTBOUNDS	在输出过程中应如何裁剪图稿作品。
	OUTPUTARTBOARDBOUNDS	
	OUTPUTCROPRECTBOUNDS	
		▶ OUTPUTARTBOUNDS — 输出尺寸是图稿的尺寸。
		▶ OUTPUTARTBOARDBOUNDS — 输出尺寸是画板的尺寸。
		▶ OUTPUTCROPRECTBOUNDS — 输出大小是作物面积的大小。
AutoCADColors		
	Max8Colors	Max256Colors
	Max16Colors	TrueColors

常量	值	注释	
AutoCADCompatibility			
	AutoCADRelease13	AutoCADRelease18	
	AutoCADRelease14	AutoCADRelease21	
	AutoCADRelease15	AutoCADRelease24	
AutoCADExportFileFormat			
	DXF	DWG	
AutoCADExportOption			
	PreserveAppearance		
	MaximizeEditability		
AutoCADGlobalScaleOption			
	OriginalSize	ScaleByValue	
	FitArtboard		
AutoCADRasterFormat			
	PNG	JPEG	
AutoCADUnit			
	Points	Millimeters	
	Picas	Centimeters	
	Inches	Pixels	
AutoKernType			
	NOAUTOKERN	OPTICAL	
	AUTO	METRICSROMANONLY	
AutoLeadingType			
	BOTTOMTOBOTTOM	TOPTOTOP	
BaselineDirectionType			
	Standard	VerticalRotated	
	TateChuYoko		
BlendAnimationType			
	INBUILD	NOBLENDANIMATION	
	INSEQUENCE		
BlendModes			
	COLORBLEND	LIGHTEN	合成对象时使用的混合模式。
	COLORBURN	LUMINOSITY	
	COLORDODGE	MULTIPLY	
	DARKEN	NORMAL	
	DIFFERENCE	OVERLAY	
	EXCLUSION	SATURATIONBLEND	
	HARDLIGHT	SCREEN	
	HUE	SOFTLIGHT	

常量	值	注释
BlendsExpandPolicy	AUTOMATICALLYCONVERTBLEND RASTERIZEBLEND	FXG文件格式用于扩展混合的策略。
BurasagariTypeEnum	Forced None	Standard
CaseChangeType	LOWERCASE SENTENCECASE	TITLECASE UPPERCASE
ColorConversion	COLORCONVERSIONREPURPOSE COLORCONVERSIONTODEST None	
ColorConvertPurpose	defaultpurpose previewpurpose	exportpurpose dummpurpose 使用Application类的ConvertSampleColor方法进行颜色转换的目的。
ColorDestination	COLORDESTINATIONDOCCMYK COLORDESTINATIONDOCRGB COLORDESTINATIONPROFILE COLORDESTINATIONWORKINGCMYK COLORDESTINATIONWORKINGRGB None	
ColorDitherMethod	DIFFUSION NOISE	NOREDUCTION PATTERNDITHER 用于抖动导出的GIF和PNG8图像中的颜色的方法。
ColorModel	PROCESS REGISTRATION	SPOT
ColorProfile	INCLUDEALLPROFILE INCLUDEDESTPROFILE INCLUDERGBPROFILE	LEAVEPROFILEUNCHANGED None
ColorReductionMethod	ADAPTIVE PERCEPTUAL	SELECTIVE WEB 用于减少导出的GIF和PNG8图像中的颜色数量的方法。

常量	值	注释
ColorType		
	CMYK GRADIENT GRAY NONE	PATTERN RGB SPOT
		单个颜色的颜色规范。
Compatibility		
	ILLUSTRATOR8 ILLUSTRATOR9 ILLUSTRATOR10 ILLUSTRATOR11 ILLUSTRATOR16 JAPANESEVERSION3	ILLUSTRATOR12 ILLUSTRATOR13 ILLUSTRATOR14 ILLUSTRATOR15 ILLUSTRATOR17 ILLUSTRATOR19
		保存EPS或Illustrator文件时要创建的Illustrator文件的版本
CompressionQuality		
	AUTOMATICJPEG2000HIGH AUTOMATICJPEG2000LOSSLESS AUTOMATICJPEG2000LOW AUTOMATICJPEG2000MAXIMUM AUTOMATICJPEG2000MEDIUM AUTOMATICJPEG2000MINIMUM AUTOMATICJPEGHIGH AUTOMATICJPEGLOW AUTOMATICJPEGMAXIMUM AUTOMATICJPEGMEDIUM AUTOMATICJPEGMINIMUM JPEG2000HIGH JPEG2000LOSSLESS	JPEG2000LOW JPEG2000MAXIMUM JPEG2000MEDIUM JPEG2000MINIMUM JPEGHIGH JPEGLOW JPEGMAXIMUM JPEGMEDIUM JPEGMINIMUM ZIP4BIT ZIP8BIT None
		保存PDF文件时使用的位图压缩质量
CoordinateSystem		
	DOCUMENTCOORDINATESYSTEM ARTBOARDCOORDINATESYSTEM	
		Illustrator使用的坐标系
CropOptions		
	Japanese Standard	
		文件裁剪框的样式
DocumentArtboardLayout		
	GridByRow GridByCol Row Column	RLGridByRow RLGridByCol RLRow
		新文档中的布局。
DocumentColorSpace		
	CMYK	RGB
		文件的色彩空间
DocumentLayoutStyle		
	CASCADE HORIZONTALTILE VERTICALTILE	FLOATALL CONSOLIDATEALL
		文档的布局样式。

常量	值	注释
DocumentPresetType		
	BasicCMYK BasicRGB Print	Mobile Video Web
		预设类型可用于新文档。
DocumentPreviewMode		
	DefaultPreview PixelPreview	OverprintPreview
		文件预览模式
DocumentRasterResolution		
	ScreenResolution MediumResolution	HighResolution
		预设文件栅格分辨率
DocumentTransparencyGrid		
	TransparencyGridNone TransparencyGridLight TransparencyGridMedium TransparencyGridDark TransparencyGridRed TransparencyGridOrange TransparencyGridGreen TransparencyGridBlue TransparencyGridPurple	
		文档透明度网格颜色
DocumentType		
	EPS ILLUSTRATOR	PDF FXG
		用于保存文件的文件格式
DownsampleMethod		
	AVERAGEDOWNSAMPLE BICUBICDOWNSAMPLE	NODOWNSAMPLE SUBSAMPLE
ElementPlacement		
	INSIDE PLACEATBEGINNING PLACEATEND	PLACEBEFORE PLACEAFTER
EPSPostScriptLevelEnum		
	LEVEL2 LEVEL3	
EPSPreview		
	BWTIFF COLORTIFF TRANSPARENTCOLORTIFF None	
		保存EPS文件时使用的预览图像格式

常量	值	注释
ExportType		
	FLASH GIF JPEG Photoshop AutoCAD	PNG24 PNG8 SVG TIFF
		用于导出文件的文件格式
FigureStyleType		
	DEFAULTFIGURESTYLE PROPORTIONAL PROPORTIONALOLDSTYLE	TABULAR TABULAROLDSTYLE
FiltersPreservePolicy		
	EXPANDFILTERS KEEPFILTERSEEDITABLE RASTERIZEFILTERS	
		筛选器保留FXG文件格式使用的策略。
FlashExportStyle		
	ASFLASHFILE LAYERSASFRAMES LAYERSASFILES	LAYERSASSYMBOLS TOFILES
		导出文件时用于转换Illustrator图像的方法
FlashExportVersion		
	FlashVersion1 FlashVersion2 FlashVersion3 FlashVersion4 FlashVersion5	FlashVersion6 FlashVersion7 FlashVersion8 FlashVersion9
		导出的SWF文件的版本
FlashImageFormat		
	LOSSLESS LOSSY	
		用于存储闪存图像的格式
FlashJPEGMethod		
	Optimized Standard	
		用于存储JPEG图像的方法
FlashPlaybackSecurity		
	PlaybackLocal PlaybackNetwork	
FontBaselineOption		
	NORMALBASELINE SUPERSCRIPIT SUBSCRIPIT	
FontCapsOption		
	ALLCAPS ALLSMALLCAPS	NORMALCAPS SMALLCAPS

常量	值	注释	
FontOpenTypePositionOption			
	DENOMINATOR	OPENTYPESUBSCRIPT	
	NUMERATOR	OPENTYPESUPERSCRIP	
	OPENTYPEDEFAULT		
FontSubstitutionPolicy			
	SUBSTITUTEDEVICE		
	SUBSTITUTEOBLIQUE		
	SUBSTITUTETINT		
FXGVersion			
	VERSION1PT0	FXG文件格式版本。	
	VERSION2PT0		
GradientsPreservePolicy			
	AUTOMATICALLYCONVERTGRADIENTS	渐变保留FXG文件格式使用的策略。	
	KEEPGRADIENTSEDTABLE		
GradientType			
	LINEAR	渐变类型	
	RADIAL		
ImageColorSpace			
	CMYK	Separation	栅格项目或导出文件的色彩空间
	Grayscale	DeviceN	
	RGB	Indexed	
	LAB		
InkPrintStatus			
	CONVERTINK		
	ENABLEINK		
	DISABLEINK		
InkType			
	BLACKINK	MAGENTAINK	
	CUSTOMINK	YELLOWINK	
	CYANINK		
JavaScriptExecutionMode			
	BeforeRunning		
	OnRuntimeError		
	never		
Justification			
	CENTER	FULLJUSTIFYLASTLINECENTER	文字的对齐方式
	LEFT	FULLJUSTIFYLASTLINELEFT	
	RIGHT	FULLJUSTIFYLASTLINERIGHT	
	FULLJUSTIFY		

常量	值	注释	
KinsokuOrderEnum			
	PUSHIN PUSHOUTONLY PUSHOUTFIRST		
KnockoutState			
	DISABLED ENABLED	INHERITED Unknown	
LanguageType			
	BOKMALNORWEGIAN BRAZILLIANPORTUGUESE BULGARIAN CANADIANFRENCH CATALAN CHINESE CZECH DANISH DUTCH DUTCH2005REFORM ENGLISH FINNISH GERMAN2006REFORM GREEK HUNGARIAN ICELANDIC ITALIAN	JAPANESE NYNORSKNORWEGIAN OLDGERMAN POLISH RUMANIAN RUSSIAN SERBIAN SPANISH STANDARDFRENCH STANDARDGERMAN STANDARDPORTUGUESE SWEDISH SWISSGERMAN SWISSGERMAN2006REFORM TURKISH UKENGLISH UKRANIAN	
LayerOrderType			
	TOPDOWN BOTTOMUP		
LibraryType			
	IllustratorArtwork Swatches Brushes	GraphicStyles Symbols	Illustrator库类型
MonochromeCompression			
	CCIT3 CCIT4 MONOZIP	None RUNLENGTH	保存PDF文件时用于单色位图项目的压缩类型
OutputFlattening			
	PRESERVEAPPEARANCE PRESERVEPATHS		保存兼容性设置为早于Illustrator 10的Illustrator版本的EPS和Illustrator文件格式时，应如何使透明度变平

常量	值	注释	
PageMarksTypes			
	Japanese		
	Roman		
PathPointSelection			
	ANCHORPOINT	选择路径的哪些点 (如果有)	
	LEFTDIRECTION		
	LEFTRIGHTPOINT		
PDFBoxType			
	PDFARTBOX	PDFCROPBOX	
	PDFBLEEDBOX	PDFMEDIABOX	
	PDFBOUNDINGBOX	PDFTRIMBOX	
PDFChangesAllowedEnum			
	CHANGE128ANYCHANGES	CHANGE40ANYCHANGES	
	CHANGE128COMMENTING	CHANGE40COMMENTING	
	CHANGE128EDITPAGE	CHANGE40PAGELAYOUT	
	CHANGE128FILLFORM	CHANGE40NONE	
	CHANGE128NONE		
PDFCompatibility			
	ACROBAT4	ACROBAT7	保存PDF文件时要创建的Acrobat文件格式的版本
	ACROBAT5	ACROBAT8	
	ACROBAT6		
PDFOverprint			
	DISCARDPDFOVERPRINT		
	PRESERVEPDFOVERPRINT		
PDFPrintAllowedEnum			
	PRINT128HIGHRESOLUTION		
	PRINT128LOWRESOLUTION		
	PRINT128NONE		
	PRINT40HIGHRESOLUTION		
	PRINT40NONE		
PDFTrimMarkWeight			
	TRIMMARKWEIGHT0125		
	TRIMMARKWEIGHT05		
	TRIMMARKWEIGHT025		
PDFXStandard			
	PDFXNONE	PDFX32002	
	PDFX1A2001	PDFX32003	
	PDFX1A2003	PDFX42007	

常量	值	注释
PerspectiveGridType		
	OnePointPerspectiveGridType	
	TwoPointPerspectiveGridType	
	ThreePointPerspectiveGridType	
	InvalidPerspectiveGridType	
PerspectiveGridPlaneType		
	GRIDLEFTPLANETYPE	
	GRIDRIGHTPLANETYPE	
	GRIDFLOORPLANETYPE	
	INVALIDGRIDPLANETYPE	
PhotoshopCompatibility		
	Photoshop6	
	Photoshop8	
PointType		
	CORNER	所选路径点的类型
	SMOOTH	
PolarityValues		
	NEGATIVE	
	POSITIVE	
PostScriptImageCompressionType		
	IMAGECOMPRESSIONNONE	
	RLE	
	JPEG	
PrintArtworkDesignation		
	ALLLAYERS	
	VISIBLELAYERS	
	VISIBLEPRINTABLELAYERS	
PrintColorIntent		
	ABSOLUTECOLORIMETRIC	
	PERCEPTUALINTENT	
	RELATIVECOLORIMETRIC	
	SATURATIONINTENT	
PrintColorProfile		
	CUSTOMPROFILE	PRINTERPROFILE
	OLDSTYLEPROFILE	SOURCEPROFILE
PrintColorSeparationMode		
	COMPOSITE	
	HOSTBASEDSEPARATION	
	INRIPSEPARATION	

常量	值	注释
PrinterColorMode		
	BLACKANDWHITEPRINTER	
	GRAYSCALEPRINTER	
	COLORPRINTER	
PrinterPostScriptLevelEnum		
	PSLEVEL1	
	PSLEVEL2	
	PSLEVEL3	
PrinterTypeEnum		
	NONPOSTSCRIPTPRINTER	
	POSTSCRIPTPRINTER	
	Unknown	
PrintFontDownloadMode		
	DOWNLOADNONE	
	DOWNLOADCOMPLETE	
	DOWNLOADSUBSET	
PrintingBounds		
	ARTBOARDBOUNDS	
	ARTWORKBOUNDS	
PrintOrientation		
	AUTOROTATE	图稿的打印方向。
	LANDSCAPE	REVERSELANDSCAPE
	PORTRAIT	REVERSEPORTRAIT
PrintPosition		
	TRANSLATEBOTTOM	TRANSLATERIGHT
	TRANSLATEBOTTOMLEFT	TRANSLATETOP
	TRANSLATEBOTTOMRIGHT	TRANSLATETOPLEFT
	TRANSLATECENTER	TRANSLATETOPRIGHT
	TRANSLATELEFT	
PrintTiling		
	TILEFULLPAGES	
	TILESINGLEFULLPAGE	
	TILEIMAGEABLEAREAS	
RasterizationColorModel		
	DEFAULTCOLORMODEL	栅格化的颜色模型。
	BITMAP	
	GRAYSCALE	
RasterLinkState		
	DATAFROMFILE	光栅项目链接图像的状态 (如果图
	DATAMODIFIED	像存储在外部)
	NODATA	

常量	值	注释	
RulerUnits			
	Centimeters	Qs	
	Inches	Pixels	
	Millimeters	Unknown	
	Picas		
	Points		
SaveOptions			
	DONOTSAVECHANGES	保存关闭文档时提供的选项	
	SAVECHANGES		
	PROMPTTOSAVECHANGES		
ScreenMode			
	DESKTOP	视图的显示方式	
	MULTIWINDOW		
	FULLSCREEN		
SpotColorKind			
	SpotCMYK	自定义颜色的专色	
	SpotLAB		
	SpotRGB		
StrokeCap			
	BUTTENDCAP	路径笔划的线上限类型	
	ROUNDENDCAP		
	PROJECTINGENDCAP		
StrokeJoin			
	BEVELENDJOIN	路径笔触的关节类型	
	ROUNDENDJOIN		
	MITERENDJOIN		
StyleRunAlignmentType			
	bottom	icfTop	
	center	ROMANBASELINE	
	icfBottom	top	
SVGCSSPropertyLocation			
	ENTITIES	STYLEATTRIBUTES	如何将文档的CSS属性包含在导出的SVG文件中
	PRESENTATIONATTRIBUTES	STYLEELEMENTS	
SVGDocumentEncoding			
	ASCII	导出SVG文件时，文档中的文本应如何编码	
	UTF8		
	UTF16		

常量	值	注释
SVGDTDVersion		
	SVG1_0	导出文件的SVB版本兼容性
	SVG1_1	
	SVGBASIC1_1	
SVGFontSubsetting		
	ALLGLYPHS	导出的SVG文件中应包含哪些字体字形
	COMMONENGLISH	
	COMMONROMAN	
	GLYPHSUSED	
SVGFontType		
	CEFFONT	导出的SVG文件中包含的字体类型
	SVGFONT	
	OUTLINEFONT	
SymbolRegistrationPoint		
	SYMBOLTOPLEFTPOINT	符号注册点
	SYMBOLTOPMIDDLEPOINT	
	SYMBOLTOPRIGHTPOINT	
	SYMBOLMIDDLELEFTPOINT	
	SYMBOLCENTERPOINT	
	SYMBOLMIDDLERIGHTPOINT	
	SYMBOLBOTTOMLEFTPOINT	
	SYMBOLBOTTOMMIDDLEPOINT	
	SYMBOLBOTTOMRIGHTPOINT	
TabStopAlignment		
	Center	制表位的对齐
	Decimal	
TextAntialias		
	CRISP	文字艺术项目中的文字抗锯齿类型
	NONE	
	SHARP	
	STRONG	
TextOrientation		
	HORIZONTAL	文字艺术作品中文字的方向
	VERTICAL	
TextPreservePolicy		
	AUTOMATICALLYCONVERTTEXT	FXG文件格式使用的文本保留策略。
	OUTLINETEXT	
	KEEPTEXTEDITABLE	
	RASTERIZETEXT	
TextType		
	AREATEXT	该对象显示的文字样式
	POINTTEXT	
	PATHTEXT	

常量	值	注释
TIFFByteOrder		
	IBMPC MACINTOSH	用于导出的TIFF文件的字节顺序。
TracingModeType		
	TRACINGMODEBLACKANDWHITE TRACINGMODECOLOR TRACINGMODEGRAY	
Transformation		
	BOTTOM LEFT BOTTOMLEFT RIGHT BOTTOMRIGHT TOP CENTER TOPLEFT DOCUMENTORIGIN TOPRIGHT	用作围绕对象旋转，调整大小或变换的锚点的点
TrappingType		
	IGNOREOPAQUE OPAQUE NORMALTRAPPING TRANSPARENT	
UserInteractionLevel		
	DISPLAYALERTS DONTDISPLAYALERTS	用户界面设置
VariableKind		
	GRAPH Unknown IMAGE VISIBILITY TEXTUAL	文档中包含什么类型的变量
ViewRasterType		
	TRACINGVIEWRASTERADJUSTEDIMAGE TRACINGVIEWRASTERNOIMAGE TRACINGVIEWRASTERORIGINALIMAGE TRACINGVIEWRASTERTRANSPARENTIMAGE	用于跟踪的栅格可视化模式。
ViewVectorType		
	TRACINGVIEWVECTORNOTRACINGRESULT TRACINGVIEWVECTOROUTLINES TRACINGVIEWVECTOROUTLINESWITHTRACING TRACINGVIEWVECTORTRACINGRESULT	跟踪的矢量可视化模式。
WariChuJustificationType		
	Center Left Right WARICHUAUTOJUSTIFY WARICHUFULLJUSTIFY WARICHUFULLJUSTIFYLASTLINECENTER WARICHUFULLJUSTIFYLASTLINELEFT WARICHUFULLJUSTIFYLASTLINERIGHT	

常量	值	注释
ZOrderMethod		
	BRINGFORWARD BRINGTOFRONT	SENDERBACKWARD SENDERTOBACK
		按照 zOrder 方法指定的用于按其父组或图层的堆叠顺序排列艺术品的位置的方法
