



**ADOBE PHOTOSHOP CC 2015**

# **JAVASCRIPT SCRIPTING REFERENCE**



© 2015 Adobe Systems Incorporated. All rights reserved.

Adobe® Photoshop® CC 2015 JavaScript Scripting Reference for Windows® and Macintosh®.

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

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

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

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

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

All other trademarks are the property of their respective owners.

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

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

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

---

# Contents

---

<b>1</b>	<b>Introduction .....</b>	<b>32</b>
	JavaScript support in Adobe Photoshop CC 2015 .....	32
	Executing scripts .....	33
	Installing scripts .....	33
	Executing other scripts .....	33
	Startup scripts .....	33
	Object model usage and naming .....	34
<b>2</b>	<b>JavaScript Object Reference .....</b>	<b>35</b>
	Object descriptions .....	35
	Properties notation .....	35
	displayDialogs .....	35
	Methods notation .....	35
	crop .....	36
	ActionDescriptor .....	37
	Properties .....	37
	count .....	37
	typename .....	37
	Methods .....	37
	clear .....	37
	erase .....	37
	fromStream .....	37
	getBoolean .....	37
	getClass .....	37
	getData .....	37
	getDouble .....	37
	getEnumerationType .....	37
	getEnumerationValue .....	37
	getInteger .....	37
	getKey .....	37
	getLargeInteger .....	38
	getList .....	38
	getObjectType .....	38
	getObjectValue .....	38
	getPath .....	38
	getReference .....	38
	getString .....	38
	getType .....	38
	getUnitDoubleType .....	38
	getUnitDoubleValue .....	38
	hasKey .....	38
	isEqual .....	38
	putBoolean .....	38
	putClass .....	38
	putData .....	38
	putDouble .....	38

putEnumerated .....	38
putInteger .....	39
putLargeInteger .....	39
putList .....	39
putObject .....	39
putPath .....	39
putReference .....	39
putString .....	39
putUnitDouble .....	39
toStream .....	39
ActionList .....	40
Properties .....	40
count .....	40
typename .....	40
Methods .....	40
clear .....	40
getBoolean .....	40
getClass .....	40
getData .....	40
getDouble .....	40
getEnumerationType .....	40
getEnumerationValue .....	40
getInteger .....	40
getLargeInteger .....	41
getList .....	41
getObjectType .....	41
getObjectValue .....	41
getPath .....	41
getReference .....	41
getString .....	41
getType .....	41
getUnitDoubleType .....	41
getUnitDoubleValue .....	41
putBoolean .....	41
putClass .....	41
putData .....	41
putDouble .....	41
putEnumerated .....	41
putInteger .....	41
putLargeInteger .....	41
putList .....	41
putObject .....	42
putPath .....	42
putReference .....	42
putString .....	42
putUnitDouble .....	42
ActionReference .....	43
Properties .....	43
typename .....	43
Methods .....	43
getContainer .....	43

getDesiredClass.....	43
getEnumeratedType.....	43
getEnumeratedValue .....	43
getForm.....	43
getIdentifier.....	43
getIndex .....	43
getName .....	43
getOffset .....	43
getProperty.....	43
putClass.....	44
putEnumerated .....	44
putIdentifier.....	44
putIndex.....	44
putName .....	44
putOffset.....	44
putProperty.....	44
Application .....	45
Properties.....	45
activeDocument.....	45
backgroundColor.....	45
build.....	45
colorSettings .....	45
currentTool .....	45
displayDialogs.....	45
documents .....	45
fonts.....	45
foregroundColor .....	45
freeMemory .....	46
locale .....	46
macintoshFileTypes .....	46
measurementLog .....	46
name.....	46
notifiers .....	46
notifiersEnabled .....	46
path.....	46
playbackDisplayDialogs .....	46
playbackParameters .....	46
preferences .....	46
preferencesFolder .....	46
recentFiles .....	46
scriptingBuildDate .....	46
scriptingVersion .....	46
systemInformation.....	46
typename .....	46
version .....	47
windowsFileTypes.....	47
Methods .....	47
batch .....	47
beep.....	47
bringToFront .....	47
changeProgressText.....	47

charIDToTypeID.....	47
doAction .....	47
doForcedProgress .....	47
doProgress .....	48
doProgressSegmentTask .....	48
doProgressSubTask.....	48
doProgressTask .....	48
eraseCustomOptions.....	48
executeAction .....	49
executeActionGet.....	49
featureEnabled .....	49
getCustomOptions.....	49
isQuicktimeAvailable.....	49
load .....	49
makeContactSheet.....	49
makePDFPresentation .....	49
makePhotoGallery.....	49
makePhotomerge.....	49
makePicturePackage .....	49
open .....	50
openDialog .....	50
purge.....	50
putCustomOptions .....	50
refresh .....	50
refreshFonts.....	50
runMenuItem .....	50
showColorPicker .....	51
stringIDToTypeID.....	51
togglePalettes.....	51
toolSupportsBrushes .....	51
(tool) .....	51
TypeIDToCharID .....	51
TypeIDToStringID .....	51
updateProgress .....	51
ArtLayer.....	54
Properties.....	54
allLocked.....	54
blendMode.....	54
bounds.....	54
boundsNoEffects .....	54
fillOpacity.....	54
filterMaskDensity .....	54
filterMaskFeather .....	54
grouped .....	54
isBackgroundLayer.....	54
kind .....	54
layerMaskDensity.....	55
layerMaskFeather .....	55
linkedLayers.....	55
name.....	55
opacity .....	55

parent.....	55
pixelsLocked .....	55
positionLocked .....	55
textItem.....	55
transparentPixelsLocked.....	55
typename .....	55
vectorMaskDensity.....	55
vectorMaskFeather .....	55
visible .....	55
xmpMetadata.....	55
Methods .....	56
adjustBrightnessContrast .....	56
adjustColorBalance .....	56
adjustCurves .....	56
adjustLevels .....	56
applyAddNoise .....	56
applyAverage .....	56
applyBlur.....	56
applyBlurMore .....	56
applyClouds.....	56
applyCustomFilter .....	56
applyDeInterlace.....	57
applyDespeckle .....	57
applyDifferenceClouds.....	57
applyDiffuseGlow .....	57
applyDisplace.....	57
applyDustAndScratches.....	57
applyGaussianBlur.....	57
applyGlassEffect.....	57
applyHighPass .....	57
applyLensBlur .....	58
applyLensFlare.....	58
applyMaximum .....	58
applyMedianNoise .....	58
applyMinimum .....	59
applyMotionBlur .....	59
applyNTSC.....	59
applyOceanRipple .....	59
applyOffset.....	59
applyPinch .....	59
applyPolarCoordinates.....	59
applyRadialBlur .....	59
applyRipple .....	59
applySharpen.....	59
applySharpenEdges.....	59
applySharpenMore .....	59
applyShear .....	60
applySmartBlur.....	60
applySpherize .....	60
applyStyle.....	60
applyTextureFill.....	60

applyTwirl .....	60
applyUnSharpMask .....	60
applyWave .....	60
applyZigZag .....	60
autoContrast .....	60
autoLevels .....	60
clear .....	61
copy .....	61
cut .....	61
desaturate .....	61
duplicate .....	61
equalize .....	61
invert .....	61
link .....	61
merge .....	61
mixChannels .....	62
move .....	62
photoFilter .....	62
posterize .....	62
rasterize .....	62
remove .....	62
resize .....	63
rotate .....	63
selectiveColor .....	63
shadowHighlight .....	63
threshold .....	63
translate .....	63
unlink .....	63
ArtLayers .....	66
Properties .....	66
length .....	66
parent .....	66
typename .....	66
Methods .....	66
add .....	66
getByName .....	66
removeAll .....	66
BatchOptions .....	67
Properties .....	67
destination .....	67
destinationFolder .....	67
errorFile .....	67
fileNameing .....	67
macintoshCompatible .....	67
overrideOpen .....	67
overrideSave .....	67
startingSerial .....	67
suppressOpen .....	67
suppressProfile .....	68
typename .....	68
unixCompatible .....	68



windowsCompatible .....	68
BitmapConversionOptions .....	69
Properties.....	69
angle.....	69
frequency.....	69
method.....	69
patternName .....	69
resolution.....	69
shape.....	69
typename .....	69
BMPSaveOptions .....	70
Properties.....	70
alphaChannels .....	70
depth.....	70
flipRowOrder .....	70
osType.....	70
rleCompression .....	70
typename .....	70
CameraRAWOpenOptions .....	71
Properties.....	71
bitsPerChannel .....	71
blueHue.....	71
blueSaturation .....	71
brightness.....	71
chromaticAberrationBY .....	71
chromaticAberrationRC.....	71
colorNoiseReduction .....	71
colorSpace.....	71
contrast .....	71
exposure .....	71
greenHue.....	71
greenSaturation .....	71
luminanceSmoothing .....	71
redHue.....	71
redSaturation .....	71
resolution.....	71
saturation .....	71
settings .....	71
shadows .....	71
shadowTint .....	71
sharpness.....	71
size .....	71
temperature.....	71
tint.....	72
typename .....	72
vignettingAmount .....	72
vignettingMidpoint .....	72
whiteBalance .....	72
Channel.....	73
Properties.....	73
color.....	73

histogram .....	73
kind .....	73
name .....	73
opacity .....	73
parent .....	73
typename .....	73
visible .....	73
Methods .....	73
duplicate .....	73
merge .....	74
remove .....	74
Channels .....	75
Properties .....	75
length .....	75
parent .....	75
typename .....	75
Methods .....	75
add .....	75
getByName .....	75
removeAll .....	75
CMYKColor .....	80
Properties .....	80
black .....	80
cyan .....	80
magenta .....	80
typename .....	80
yellow .....	80
ColorSampler .....	81
Properties .....	81
color .....	81
position .....	81
parent .....	81
typename .....	81
Methods .....	81
move .....	81
remove .....	81
ColorSamplers .....	82
Properties .....	82
length .....	82
parent .....	82
typename .....	82
Methods .....	82
add .....	82
removeAll .....	82
ContactSheetOptions .....	83
Properties .....	83
acrossFirst .....	83
bestFit .....	83
caption .....	83
columnCount .....	83
flatten .....	83

font.....	83
fontSize.....	83
height.....	83
horizontal .....	83
mode.....	83
resolution.....	83
rowCount.....	83
typename .....	83
useAutoSpacing.....	83
vertical .....	83
width .....	83
CountItem .....	84
Properties.....	84
position .....	84
parent.....	84
typename .....	84
Methods .....	84
remove.....	84
CountItems .....	85
Properties.....	85
length.....	85
parent.....	85
typename .....	85
Methods .....	85
add .....	85
getByName .....	85
removeAll .....	85
DCS1_SaveOptions.....	86
Properties.....	86
dCS .....	86
embedColorProfile.....	86
encoding.....	86
halftoneScreen .....	86
interpolation.....	86
preview.....	86
transferFunction.....	86
typename .....	86
vectorData.....	86
DCS2_SaveOptions.....	87
Properties.....	87
dCS .....	87
embedColorProfile.....	87
encoding.....	87
halftoneScreen .....	87
interpolation.....	87
multiFileDCS .....	87
preview.....	87
spotColors .....	87
transferFunction.....	87
typename .....	87
vectorData.....	87

DICOMOpenOptions .....	88
Properties.....	88
anonymize.....	88
columns.....	88
reverse .....	88
rows .....	88
showOverlays.....	88
typename .....	88
windowLevel .....	88
windowWidth .....	88
Document .....	89
Properties.....	89
activeChannels .....	89
activeHistoryBrushSource .....	89
activeHistoryState .....	89
activeLayer .....	89
artLayers.....	89
backgroundLayer.....	89
bitsPerChannel .....	89
channels .....	89
colorProfileName .....	89
colorProfileType .....	89
colorSamplers .....	89
componentChannels.....	89
countItems .....	90
fullName.....	90
guides .....	90
height.....	90
histogram .....	90
historyStates .....	90
info .....	90
layerComps .....	90
layers .....	90
layerSets.....	90
managed.....	90
measurementScale .....	90
mode .....	90
name.....	90
parent.....	90
path.....	90
pathItems .....	90
pixelAspectRatio .....	90
printSettings.....	90
quickMaskMode .....	91
resolution.....	91
saved .....	91
selection.....	91
typename .....	91
width .....	91
xmpMetadata.....	91
Methods .....	92

autoCount .....	92
changeMode .....	92
close .....	92
convertProfile .....	92
crop .....	92
duplicate .....	92
exportDocument .....	93
flatten .....	93
flipCanvas .....	93
importAnnotations .....	93
mergeVisibleLayers .....	93
paste .....	93
print .....	93
printOneCopy .....	93
rasterizeAllLayers .....	93
recordMeasurements .....	93
resizeCanvas .....	93
resizeImage .....	94
revealAll .....	94
rotateCanvas .....	94
save .....	94
saveAs .....	94
splitChannels .....	94
suspendHistory .....	94
trap .....	94
trim .....	95
DocumentPrintSettings .....	97
Properties .....	97
backgroundColor .....	97
bleedWidth .....	97
caption .....	97
centerCropMarks .....	97
colorBars .....	97
copies .....	97
cornerCropMarks .....	97
colorHandling .....	97
activePrinter .....	97
flip .....	97
hardProof .....	97
interpolate .....	97
labels .....	97
mapBlack .....	97
negative .....	97
renderIntent .....	97
posX .....	97
posY .....	97
printBorder .....	97
printerName .....	97
printSpace .....	97
registrationMarks .....	97
scale .....	98

vectorData.....	98
Methods .....	99
setPagePosition.....	99
DocumentInfo .....	100
Properties.....	100
author .....	100
authorPosition .....	100
caption.....	100
captionWriter .....	100
category .....	100
city.....	100
copyrighted .....	100
copyrightNotice .....	100
country .....	100
creationDate.....	100
credit .....	100
exif.....	100
headline .....	100
instructions .....	100
jobName .....	100
keywords.....	100
ownerUrl .....	100
parent.....	101
provinceState .....	101
source.....	101
supplementalCategories.....	101
title .....	101
transmissionReference .....	101
typename .....	101
urgency .....	101
Documents .....	103
Properties.....	103
length.....	103
parent.....	103
typename .....	103
Methods .....	103
add .....	103
getByName .....	103
EPSSaveOptions .....	104
Properties.....	104
antiAlias.....	104
constrainProportions .....	104
height.....	104
mode .....	104
resolution.....	104
typename .....	104
width .....	104
EPSSaveOptions.....	105
Properties.....	105
embedColorProfile.....	105
encoding.....	105

halftoneScreen .....	105
interpolation.....	105
preview.....	105
psColorManagement .....	105
transferFunction.....	105
transparentWhites.....	105
typename .....	105
vectorData.....	105
ExportOptionsIllustrator .....	106
Properties.....	106
path.....	106
pathName.....	106
typename .....	106
ExportOptionsSaveForWeb .....	107
Properties.....	107
blur.....	107
colorReduction .....	107
colors.....	107
dither.....	107
ditherAmount .....	107
format .....	107
includeProfile .....	107
interlaced.....	107
lossy .....	107
matteColor .....	107
optimized .....	107
PNG8.....	107
quality .....	108
transparency.....	108
transparencyAmount.....	108
transparencyDither .....	108
typename .....	108
webSnap .....	108
File.....	109
Folder.....	109
GalleryBannerOptions .....	110
Properties.....	110
contactInfo .....	110
date.....	110
font.....	110
fontSize.....	110
photographer.....	110
siteName .....	110
typename .....	110
GalleryCustomColorOptions .....	111
Properties.....	111
activeLinkColor.....	111
backgroundColor.....	111
bannerColor.....	111
linkColor .....	111
textColor .....	111

typename .....	111
visitedLinkColor .....	111
GalleryImagesOptions .....	112
Properties .....	112
border .....	112
caption .....	112
dimension .....	112
font .....	112
fontSize .....	112
imageQuality .....	112
includeCopyright .....	112
includeCredits .....	112
includeFilename .....	112
includeTitle .....	112
numericLinks .....	112
resizeConstraint .....	113
resizeImages .....	113
typename .....	113
GalleryOptions .....	114
Properties .....	114
addSizeAttributes .....	114
bannerOptions .....	114
customColorOptions .....	114
emailAddress .....	114
imagesOptions .....	114
includeSubFolders .....	114
layoutStyle .....	114
preserveAllMetadata .....	114
securityOptions .....	114
thumbnailOptions .....	114
typename .....	114
useShortExtension .....	114
useUTF8Encoding .....	114
GallerySecurityOptions .....	115
Properties .....	115
content .....	115
font .....	115
fontSize .....	115
opacity .....	115
text .....	115
textColor .....	115
textPosition .....	115
textRotate .....	115
typename .....	115
GalleryThumbnailOptions .....	116
Properties .....	116
border .....	116
caption .....	116
columnCount .....	116
dimension .....	116
font .....	116



fontSize.....	116
includeCopyright.....	116
includeCredits.....	116
includeFilename.....	116
includeTitle .....	116
rowCount.....	116
size .....	116
typename .....	116
GIFSaveOptions.....	117
Properties.....	117
colors.....	117
dither.....	117
ditherAmount .....	117
forced .....	117
interlaced.....	117
matte.....	117
palette.....	117
preserveExactColors .....	117
transparency.....	117
typename .....	117
GrayColor.....	118
Properties.....	118
gray.....	118
typename .....	118
Guide.....	119
Properties.....	119
direction .....	119
coordinate .....	119
Guides.....	120
Properties.....	120
length.....	120
parent.....	120
typename .....	120
Methods .....	120
add .....	120
getByName .....	120
HistoryState .....	121
Properties.....	121
name.....	121
parent.....	121
snapshot .....	121
typename .....	121
HistoryStates .....	122
Properties.....	122
length.....	122
parent.....	122
typename .....	122
Methods .....	122
getByName .....	122
HSBColor.....	123
Properties.....	123

brightness.....	123
hue .....	123
saturation .....	123
typename .....	123
IndexedConversionOptions .....	124
Properties.....	124
colors.....	124
dither.....	124
ditherAmount .....	124
forced .....	124
matte.....	124
palette.....	124
preserveExactColors .....	124
transparency.....	124
typename .....	124
JPEGSaveOptions .....	125
Properties.....	125
embedColorProfile.....	125
formatOptions .....	125
matte.....	125
quality .....	125
scans .....	125
typename .....	125
LabColor .....	126
Properties.....	126
a.....	126
b .....	126
l.....	126
typename .....	126
LayerComp.....	127
Properties.....	127
appearance .....	127
comment .....	127
name.....	127
parent.....	127
position .....	127
selected .....	127
typename .....	127
visibility .....	127
Methods .....	127
apply.....	127
recapture .....	127
remove.....	127
resetfromComp .....	127
LayerComps.....	128
Properties.....	128
length.....	128
parent.....	128
typename .....	128
Methods .....	128
add .....	128

getByName .....	128
removeAll .....	128
Layers .....	129
Properties .....	129
length .....	129
parent .....	129
typename .....	129
Methods .....	129
getByName .....	129
removeAll .....	129
LayerSet .....	130
Properties .....	130
allLocked .....	130
artLayers .....	130
blendMode .....	130
bounds .....	130
enabledChannels .....	130
layers .....	130
layerSets .....	130
linkedLayers .....	130
name .....	130
opacity .....	130
parent .....	130
typename .....	130
visible .....	130
Methods .....	131
duplicate .....	131
link .....	131
merge .....	131
move .....	131
remove .....	131
resize .....	131
rotate .....	131
translate .....	131
unlink .....	131
LayerSets .....	132
Properties .....	132
length .....	132
parent .....	132
typename .....	132
Methods .....	132
add .....	132
getByName .....	132
removeAll .....	132
MeasurementLog .....	134
Methods .....	134
exportMeasurements .....	134
deleteMeasurements .....	134
MeasurementScale .....	135
Properties .....	135
pixelLength .....	135

logicalLength .....	135
logicalUnits .....	135
NoColor .....	136
Properties.....	136
typename .....	136
Notifier.....	137
Properties.....	137
event.....	137
eventClass .....	137
eventFile .....	137
parent.....	137
typename .....	137
Methods .....	137
remove.....	137
Notifiers.....	138
Properties.....	138
length.....	138
parent.....	138
typename .....	138
Methods .....	138
add .....	138
removeAll .....	138
PathItem .....	139
Properties.....	139
kind .....	139
name.....	139
parent.....	139
subPathItems .....	139
typename .....	139
Methods .....	139
deselect .....	139
duplicate .....	139
fillPath .....	139
makeClippingPath.....	140
makeSelection .....	140
remove.....	140
select .....	140
strokePath .....	140
PathItems .....	143
Properties.....	143
length.....	143
parent.....	143
typename .....	143
Methods .....	143
add .....	143
getByName .....	143
removeAll .....	143
PathPoint.....	144
Properties.....	144
anchor .....	144
kind .....	144

leftDirection.....	144
parent.....	144
rightDirection.....	144
typename .....	144
PathPointInfo .....	145
Properties.....	145
anchor .....	145
kind .....	145
leftDirection.....	145
rightDirection.....	145
typename .....	145
PathPoints.....	147
Properties.....	147
length.....	147
parent.....	147
typename .....	147
PDFOpenOptions .....	148
Properties.....	148
antiAlias.....	148
bitsPerChannel .....	148
constrainProportions .....	148
cropPage.....	148
height.....	148
mode.....	148
name.....	148
object .....	148
page.....	148
resolution.....	148
suppressWarnings .....	148
typename .....	148
use3DObjectNumber .....	148
usePageNumber .....	148
width .....	148
PDFSaveOptions .....	149
Properties.....	149
alphaChannels .....	149
annotations.....	149
colorConversion.....	149
convertToEightBit.....	149
description .....	149
destinationProfile .....	149
downgradeColorProfile .....	149
downSample .....	149
downSampleSize .....	149
downSampleSizeLimit .....	149
embedColorProfile.....	149
embedFonts .....	149
embedThumbnail.....	149
encoding.....	149
interpolation.....	149
jpegQuality .....	150

layers .....	150
optimizeForWeb .....	150
outputCondition .....	150
outputConditionID.....	150
PDFCompatibility.....	150
PDFStandard .....	150
preserveEditing .....	150
presetFile .....	150
profileInclusionPolicy.....	150
registryName.....	150
spotColors .....	150
tileSize.....	150
transparency.....	150
typename .....	150
useOutlines .....	150
vectorData.....	151
view.....	151
PhotoCDOpenOptions .....	152
Properties.....	152
colorProfileName.....	152
colorSpace.....	152
orientation.....	152
pixelSize .....	152
resolution.....	152
typename .....	152
PhotoshopSaveOptions .....	153
Properties.....	153
alphaChannels .....	153
annotations.....	153
embedColorProfile.....	153
layers .....	153
spotColors .....	153
typename .....	153
PICTFileSaveOptions .....	154
Properties.....	154
alphaChannels .....	154
compression .....	154
embedColorProfile.....	154
resolution.....	154
typename .....	154
PICTResourceSaveOptions.....	155
Properties.....	155
alphaChannels .....	155
compression .....	155
embedColorProfile.....	155
name.....	155
resolution.....	155
resourceID .....	155
typename .....	155
PicturePackageOptions.....	156
Properties.....	156

content .....	156
flatten .....	156
font .....	156
fontSize .....	156
layout .....	156
mode .....	156
opacity .....	156
resolution .....	156
text .....	156
textColor .....	156
textPosition .....	156
textRotate .....	156
typename .....	156
PixarSaveOptions .....	157
Properties .....	157
alphaChannels .....	157
typename .....	157
PNGSaveOptions .....	158
Properties .....	158
compression .....	158
interlaced .....	158
typename .....	158
Preferences .....	159
Properties .....	159
additionalPluginFolder .....	159
appendExtension .....	159
askBeforeSavingLayeredTIFF .....	159
autoUpdateOpenDocuments .....	159
beepWhenDone .....	159
colorChannelsInColor .....	159
colorPicker .....	159
columnGutter .....	159
columnWidth .....	159
createFirstSnapshot .....	159
dynamicColorSliders .....	159
editLogItems .....	159
exportClipboard .....	160
fontPreviewSize .....	160
fullSizePreview .....	160
gamutWarningOpacity .....	160
gridSize .....	160
gridStyle .....	160
gridSubDivisions .....	160
guideStyle .....	160
iconPreview .....	160
imageCacheLevels .....	160
imagePreviews .....	160
interpolation .....	160
keyboardZoomResizesWindows .....	160
macOSThumbnail .....	160
maximizeCompatibility .....	160

maxRAMuse .....	160
nonLinearHistory .....	160
numberOfHistoryStates .....	161
otherCursors .....	161
paintingCursors .....	161
parent .....	161
pixelDoubling .....	161
pointSize .....	161
recentFileListLength .....	161
rulerUnits .....	161
saveLogItems .....	161
saveLogItemsFile .....	161
savePaletteLocations .....	161
showAsianTextOptions .....	161
showEnglishFontNames .....	161
showSliceNumber .....	161
showToolTips .....	161
smartQuotes .....	161
textFontSize .....	161
typename .....	161
typeUnits .....	162
useAdditionalPluginFolder .....	162
useHistoryLog .....	162
useLowerCaseExtension .....	162
useShiftKeyForToolSwitch .....	162
useVideoAlpha .....	162
windowsThumbnail .....	162
PresentationOptions .....	163
Properties .....	163
autoAdvance .....	163
includeFilename .....	163
interval .....	163
loop .....	163
magnification .....	163
PDFFileOptions .....	163
presentation .....	163
transition .....	163
typename .....	163
RawFormatOpenOptions .....	164
Properties .....	164
bitsPerChannel .....	164
byteOrder .....	164
channelNumber .....	164
headerSize .....	164
height .....	164
interleaveChannels .....	164
retainHeader .....	164
typename .....	164
width .....	164
RawSaveOptions .....	165
Properties .....	165



alphaChannels .....	165
spotColors .....	165
typename .....	165
RGBColor.....	166
Properties.....	166
blue .....	166
green .....	166
hexValue .....	166
red .....	166
typename .....	166
Selection .....	167
Properties.....	167
bounds.....	167
parent.....	167
solid .....	167
typename .....	167
Methods .....	167
clear .....	167
contract .....	167
copy .....	167
cut.....	167
deselect .....	167
expand.....	167
feather.....	167
fill .....	168
grow .....	168
invert .....	168
load .....	168
makeWorkPath .....	168
resize .....	168
resizeBoundary .....	168
rotate.....	168
rotateBoundary .....	168
select .....	168
selectAll .....	168
selectBorder.....	169
similar .....	169
smooth .....	169
store.....	169
stroke.....	169
translate .....	169
translateBoundary .....	169
SGIRGBSaveOptions .....	172
Properties.....	172
alphaChannels .....	172
spotColors .....	172
typename .....	172
SolidColor.....	173
Properties.....	173
cmyk .....	173
gray .....	173

hsb.....	173
lab.....	173
model.....	173
nearestWebColor.....	173
rgb.....	173
typename .....	173
Methods .....	173
isEqual.....	173
SubPathInfo.....	174
Properties.....	174
closed.....	174
entireSubPath .....	174
operation .....	174
typename .....	174
SubPathItem.....	175
Properties.....	175
closed.....	175
operation .....	175
parent.....	175
pathPoints.....	175
typename .....	175
SubPathItems.....	176
Properties.....	176
length.....	176
parent.....	176
typename .....	176
TargaSaveOptions.....	177
Properties.....	177
alphaChannels.....	177
resolution.....	177
rleCompression .....	177
typename .....	177
TextFont.....	178
Properties.....	178
family.....	178
name.....	178
parent.....	178
postScriptName .....	178
style.....	178
typename .....	178
TextFonts.....	179
Properties.....	179
length.....	179
parent.....	179
typename .....	179
Methods .....	179
getByName .....	179
TextItem.....	180
Properties.....	180
alternateLigatures .....	180
antiAliasMethod.....	180

autoKerning .....	180
autoLeadingAmount .....	180
baselineShift .....	180
capitalization .....	180
color .....	180
contents .....	180
desiredGlyphScaling .....	180
desiredLetterScaling .....	181
desiredWordScaling .....	181
direction .....	181
fauxBold .....	181
fauxItalic .....	181
firstLineIndent .....	181
font .....	181
hangingPunctuation .....	181
height .....	181
horizontalScale .....	181
hyphenateAfterFirst .....	182
hyphenateBeforeLast .....	182
hyphenateCapitalWords .....	182
hyphenateWordsLongerThan .....	182
hyphenation .....	182
hyphenationZone .....	182
hyphenLimit .....	182
justification .....	182
kind .....	182
language .....	182
leading .....	182
leftIndent .....	182
ligatures .....	182
maximumGlyphScaling .....	182
maximumLetterScaling .....	183
maximumWordScaling .....	183
minimumGlyphScaling .....	183
minimumLetterScaling .....	184
minimumWordScaling .....	184
noBreak .....	184
oldStyle .....	184
parent .....	184
position .....	184
rightIndent .....	184
size .....	184
spaceAfter .....	184
spaceBefore .....	185
strikeThru .....	185
textComposer .....	185
tracking .....	185
typename .....	185
underline .....	185
useAutoLeading .....	185
verticalScale .....	185

warpBend .....	185
warpDirection .....	185
warpHorizontalDistortion.....	185
warpStyle.....	185
warpVerticalDistortion.....	185
width .....	185
Methods .....	186
convertToShape.....	186
createPath .....	186
TiffSaveOptions.....	187
Properties.....	187
alphaChannels .....	187
annotations.....	187
byteOrder .....	187
embedColorProfile.....	187
imageCompression.....	187
interleaveChannels .....	187
jpegQuality .....	187
layerCompression.....	187
layers .....	187
saveImagePyramid.....	187
spotColors .....	187
transparency.....	187
typename .....	187
UnitValue .....	188
xmpMetadata .....	189
Properties.....	189
parent.....	189
rawData .....	189
typename .....	189
<b>3 JavaScript Resource.....</b>	<b>190</b>
JavaScript resource syntax.....	190
Basic JavaScript resource example .....	191
Enable-info grammar .....	191
Undefined values in enable-info evaluation .....	194
Using the "in" function .....	194
Action Manager automation .....	194
Terminology dictionary .....	194
Value type definitions .....	195
Uniqueness rules for terminology entries.....	196
Terminology definition example.....	196
<b>4 Scripting Constants .....</b>	<b>198</b>
AdjustmentReference .....	198
AnchorPosition .....	198
AntiAlias .....	198
AutoKernType .....	198
BatchDestinationType .....	198
BitmapConversionType.....	199
BitmapHalfToneType .....	199
BitsPerChannelType.....	199

BlendMode.....	199
BMPDepthType .....	199
ByteOrder .....	200
CameraRAWSettingsType.....	200
CameraRAWSize.....	200
ChangeMode.....	200
ChannelType .....	200
ColorBlendMode .....	201
ColorModel .....	201
ColorPicker .....	201
ColorProfileType .....	201
ColorReductionType.....	201
ColorSpaceType .....	201
CopyrightedType.....	201
CreateFields .....	202
CropToType .....	202
DCSType.....	202
DepthMapSource .....	202
DescValueType .....	202
DialogModes .....	202
Direction .....	202
DisplacementMapType .....	203
Dither .....	203
DocPositionStyle.....	203
DocumentFill.....	203
DocumentMode.....	203
EditLogItemsType.....	203
ElementPlacement.....	203
EliminateFields .....	203
ExportType.....	203
Extension .....	203
FileNamingType .....	204
FontPreviewType.....	204
FontSize.....	204
ForcedColors .....	204
FormatOptions .....	204
GalleryConstrainType.....	204
GalleryFontType.....	205
GallerySecurityTextColorType .....	205
GallerySecurityTextPositionType .....	205
GallerySecurityTextRotateType .....	205
GallerySecurityType .....	205
GalleryThumbSizeType.....	205
Geometry .....	205
GridLineStyle .....	205
GridSize .....	205
GuideLineStyle.....	206
IllustratorPathType.....	206
Intent.....	206
Justification.....	206
Language.....	206

LayerCompression.....	206
LayerKind.....	207
LensType.....	207
MagnificationType .....	207
MatteType .....	207
MeasurementRange .....	207
MeasurementSource .....	207
NewDocumentMode.....	207
NoiseDistribution.....	208
OffsetUndefinedAreas .....	208
OpenDocumentMode.....	208
OpenDocumentType.....	208
OperatingSystem.....	208
Orientation .....	208
OtherPaintingCursors .....	209
PaintingCursors .....	209
PaletteType .....	209
PathKind.....	209
PDFCompatibility.....	209
PDFEncoding.....	209
PDFResample .....	210
PDFStandard .....	210
PhotoCDColorSpace .....	210
PhotoCDSize.....	210
PICTBitsPerPixels.....	210
PICTCompression.....	210
PicturePackageTextType.....	210
PointKind .....	211
PointType .....	211
PolarConversionType .....	211
Preview .....	211
PrintColorHandling .....	211
PurgeTarget.....	211
QueryStateType.....	211
RadialBlurMethod.....	211
RadialBlurQuality .....	211
RasterizeType .....	211
ReferenceFormType .....	211
ResampleMethod .....	212
RippleSize .....	212
SaveBehavior.....	212
SaveDocumentType .....	212
SaveEncoding .....	212
SaveLogItemsType.....	212
SaveOptions .....	212
SelectionType.....	213
ShapeOperation.....	213
SmartBlurMode .....	213
SmartBlurQuality.....	213
SourceSpaceType .....	213
SpherizeMode.....	213

StrikeThruType .....	213
StrokeLocation.....	213
TargaBitsPerPixels .....	213
TextCase.....	214
TextComposer.....	214
TextType .....	214
TextureType.....	214
TIFFEncoding.....	214
ToolType .....	214
TransitionType.....	214
TrimType.....	215
TypeUnits.....	215
UndefinedAreas .....	215
UnderlineType .....	215
Units.....	215
Urgency .....	215
WarpStyle .....	215
WaveType .....	216
WhiteBalanceType.....	216
ZigZagType .....	216
<b>Appendix A: Event ID Codes .....</b>	<b>217</b>
<b>Index .....</b>	<b>225</b>

This reference describes the objects and methods in the Adobe® Photoshop® CC 2015® JavaScript™ type library. A companion document, *Photoshop CC 2015 Scripting Guide*, describes basic scripting concepts and the Photoshop object model. This document provides reference details of the Photoshop object model, and additional information on JavaScript-specific features.

Adobe Photoshop CC 2015 uses ExtendScript, Adobe's extended implementation of JavaScript. See [JavaScript support in Adobe Photoshop CC 2015](#) for additional information.

This book contains the following sections:

- This introduction, which describes scripting support in Adobe Photoshop CC 2015, and lists changes to the JavaScript interface since the previous release.
- [JavaScript Object Reference](#), which provides a complete reference for all Photoshop DOM objects and commands.
- [Scripting Constants](#), which lists all enumerations used in the Photoshop type library.

## JavaScript support in Adobe Photoshop CC 2015

For a JavaScript file to be recognized by Photoshop as a valid script file, it must use either a `.js` or a `.jsx` extension.

On the Mac OS, there is no difference in the way scripts with the two extensions function. On Windows, if the script files is opened from inside Photoshop, there is no difference between using the `.js` and `.jsx` extension. However, if the script is launched by double-clicking on it, a script with the `.js` extension is interpreted with the Microsoft JScript engine, and it cannot launch Adobe Photoshop CC 2015. For Windows, using the `.jsx` extension is preferable, since it interprets the script with the ExtendScript engine.

All of the Adobe Creative Cloud applications, including Adobe Photoshop CC 2015, use ExtendScript, Adobe's extended implementation of JavaScript. ExtendScript files are distinguished by the `.jsx` extension. ExtendScript offers all standard JavaScript features, plus additional features and utilities, such as:

- A debugging environment (the ExtendScript Toolkit)
- A localization utility
- Tools that allow you to combine scripts and direct them to particular applications
- Platform-independent file and folder representation

Many of the JavaScript objects and methods use objects defined in ExtendScript, such as the `File` object, the `Folder` object, and the `UnitValue` object. For that reason, using the `.jsx` extension for your script files is preferable.

For details of these and additional features, see the *JavaScript Tools Guide*. This document is installed with Creative Cloud applications at these locations:

- In Windows:

C:\Program Files\Adobe\Adobe Utilities\ExtendScript Toolkit CC 2015\SDK



- In Mac OS:

Applications/Utilities/Adobe Utilities/ExtendScript Toolkit CC/SDK

The latest versions of this document and of the ExtendScript Toolkit, can also be downloaded from Adobe Developer Center, <http://www.adobe.com/devnet/>.

## Executing scripts

The Adobe Photoshop CC 2015 interface includes a Scripts menu (**File > Scripts**) which provides quick and easy access to your JavaScripts. Scripts can be listed directly as menu items that run when you select them, or you can navigate to and run any JavaScript in your file system.

If Adobe Photoshop CC 2015 encounters an error during script execution, it displays the error message.

## Installing scripts

To install a JavaScript in the Scripts menu, place it in the Scripts folder (**Photoshop CC 2015/Presets/Scripts**). The names of the scripts in the Scripts folder, without the file name extension, will be displayed in the Scripts menu. Any number of scripts may be installed in the Scripts menu.

Scripts added to the Scripts folder while Adobe Photoshop CC 2015 is running will not appear in the Scripts menu until the next time you launch the application.

All scripts found in the Scripts folder and sub-folders are displayed at the top level of the **File > Scripts** menu. The addition of sub-folders does not add a hierarchical organization to the Scripts menu.

## Executing other scripts

The **Browse** item at the end of the **Scripts** menu (**File > Scripts > Browse**) allows you to execute scripts which are not installed in the Scripts folder. You can also use Browse to select scripts installed in the Scripts folder after the application was last launched.

Selecting **Browse** displays a file browser dialog which allows you to select a script file for execution. Only `.js` or `.jsx` files are displayed in the browse dialog. When you select a script file, it is executed the same way as an installed script.

## Startup scripts

On startup, Adobe Photoshop CC 2015 executes all `.jsx` files that it finds in the startup folders.

- On Windows, the startup folder for user-defined scripts is:

C:\Program Files\Common Files\Adobe\Startup Scripts CC\Adobe Photoshop

- On Mac OS, the startup folder for user-defined scripts is:

~/Library/Application Support/Adobe/Startup Scripts CC/Adobe Photoshop

If a script is meant to be executed only by Adobe Photoshop CC 2015, it must include code such as the following:

```
if ( BridgeTalk.appName == "photoshop" ) {  
    //continue executing script  
}
```

For additional details, see the *JavaScript Tools Guide*.

## Object model usage and naming

The JavaScript API follows JavaScript naming conventions in that all classes (object types) begin with uppercase letters and have mixed case. Typically, in JavaScript, you instantiate classes using the `new` operator:

```
new ClassName();
```

However, in the Photoshop Object Model, it is often not necessary to do this. Major object types are collected into collection classes; for example, a list of `Document` objects is contained in a `Documents` collection object. You then access the collection object through a corresponding collection property in its container in the object hierarchy.

For example, the collection of all open documents is contained in the top-level `Application` object. You can access this through the global variable `app`, or simply reference its properties directly at the top level:

```
app.documents[0] // get the first loaded documented
documents[0] // this is the same
```

A collection property has the same name as the collection object, but begins with lowercase. For example, a `Document` contains a collection of `LayerSets`, and a `LayerSet` contains a collection of `ArtLayers`. To access one `ArtLayer` object in a set:

```
var myLayer = activeDocument.layerSets[0].artLayers[0];
```

The collections, as in this example, can be treated as arrays, which is useful for iteration. They also provide methods to create their contained objects, and to access them by name:

```
var newLayer = activeDocument.artLayers.add(); // Create a new ArtLayer object
newLayer.name = "My Layer"; // name it for later reference
...
var layerRef = activeDocument.artLayers.getByName("My Layer");
```

Some objects, such as the `Font` objects contained in the `app.fonts` collection, are created by the application, and never by your scripts.

Your scripts do use the JavaScript `new` operator to create helper objects, such as those that encapsulate a set of options for opening or saving a document in a particular format:

```
var opts = new PDFOpenOptions();
opts.page = 10;
app.open(myPDFFile, opts);
```

The Photoshop objects (the JavaScript type library for Adobe® Photoshop® CC 2015) are presented alphabetically and in tabular format in this chapter. Sample code for several object model classes is given to help illustrate the syntax as well as usage of the object class.

## Object descriptions

Object properties and methods are described in separate tables for each object. The following sections describe the conventions used in these descriptions.

### Properties notation

The Properties table for an object lists the following:

- The properties defined in each object.
- The value type for each property.

When the value type is a constant or another object, the value is a hypertext link to the listing for that constant or object.

- The property's input status: read-only or read-write.
- A description that explains what the property does.

Property	Value type	What it is
<code>displayDialogs</code>	<a href="#">DialogModes</a>	Read-write. The dialog mode for the application, which controls what types of dialogs should be displayed when running scripts.

For constants, like [DialogModes](#) in the sample, click the link to go to the table that shows allowed values. Constants are represented by objects, and allowed values are properties of those objects. Specify a constant value in the form *ConstantName.VALUE*. For example:

```
app.displayDialogs = DialogModes.ERROR;
```

### Methods notation

The Methods table for an object lists the following:

- The method name.
- The parameters list.
- The parameter value types, on lines corresponding to each parameter.
- Return value type
- A description of what the method does, and further descriptions of parameters, if needed.

Method	Parameter type	Returns	What it does
<b>crop</b> (bounds [, angle] [, width] [, height])	array of 4 <a href="#">UnitValue</a> number <a href="#">UnitValue</a> <a href="#">UnitValue</a>		Crops the document. The <code>bounds</code> parameter is an array of four coordinates for the region remaining after cropping, [left, top, right, bottom].

When a parameter type or return value is a constant or another object, the value is a hypertext link to the listing for that constant or object.

Parameters can be required or optional. Optional parameters are indicated in the table by square brackets ([ ]). In the example, the first parameter, *bounds*, is required. The remaining parameters are all optional.

You must pass a value for each required parameter. You can leave out optional parameters if there are no remaining values to pass; however, if you wish to use the default value for any optional parameter that is not the last one specified, pass `undefined` as a placeholder. You must enter the values in the order they are listed, so that the JavaScript compiler knows which value you are entering.

For example, the following passes only the required parameter (using a previously-defined variable for the bounding region):

```
app.activeDocument.crop( myRegion );
```

The following skips the *angle* parameter, specifies the *width* value, and omits the final *height* value:

```
var myWidth = new UnitValue( "500 pixels" );  
app.executeAction( myRegion, undefined, myWidth );
```

# ActionDescriptor

This object provides a dictionary-style mechanism for storing data as key-value pairs. It can be used for low-level access into Photoshop. See an example of this usage in [‘Selection sample script’ on page 169](#).

Many configuration files use serialized action descriptors to represent their data. It is used, for example, to encapsulate playback options in [Application.playbackParameters](#), and is returned by [Application.getCustomOptions\(\)](#).

## Properties

Property	Value type	What it is
<b>count</b>	number	Read-only. The number of keys contained in the descriptor.
<b>typename</b>	string	Read-only. The class name of the referenced <code>actionDescriptor</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>clear</b> ( )			Clears the descriptor.
<b>erase</b> (key)	number		Erases a key from the descriptor.
<b>fromStream</b> (value)	string		Creates a descriptor from a stream of bytes; for reading from disk.
<b>getBoolean</b> (key)	number	boolean	Gets the value of a key of type boolean.
<b>getClass</b> (key)	number	number	Gets the value of a key of type class.
<b>getData</b> (key)	number	string	Gets raw byte data as a string value.
<b>getDouble</b> (key)	number	number	Gets the value of a key of type double.
<b>getEnumerationType</b> (key)	number	number	Gets the enumeration type of a key.
<b>getEnumerationValue</b> (key)	number	number	Gets the enumeration value of a key.
<b>getInteger</b> (key)	number	number	Gets the value of a key of type integer.
<b>getKey</b> (index)	number	number	Gets the ID of the <i>N</i> th key, provided by index.

Method	Parameter type	Returns	What it does (Continued)
<b>getLargeInteger</b> (key)	number	number	Gets the value of a key of type large integer.
<b>getList</b> (key)	number	<a href="#">ActionList</a>	Gets the value of a key of type list.
<b>getObjectType</b> (key)	number	number	Gets the class ID of an object in a key of type object.
<b>getObjectValue</b> (key)	number	<a href="#">ActionDescriptor</a>	Gets the value of a key of type object.
<b>getPath</b> (key)	number	<a href="#">File</a>	Gets the value of a key of type File.
<b>getReference</b> (key)	number	<a href="#">ActionReference</a>	Gets the value of a key of type <a href="#">ActionReference</a> .
<b>getString</b> (key)	number	string	Gets the value of a key of type string.
<b>getType</b> (key)	number	<a href="#">DescValueType</a>	Gets the type of a key.
<b>getUnitDoubleType</b> (key)	number	number	Gets the unit type of a key of type UnitDouble.
<b>getUnitDoubleValue</b> (key)	number	number	Gets the value of a key of type UnitDouble.
<b>hasKey</b> (key)	number	boolean	Checks whether the descriptor contains the provided key.
<b>isEqual</b> (otherDesc)	<a href="#">ActionDescriptor</a>	boolean	Determines whether the descriptor is the same as another descriptor.
<b>putBoolean</b> (key, value)	number boolean		Sets the value for a key whose type is boolean.
<b>putClass</b> (key, value)	number number		Sets the value for a key whose type is class.
<b>putData</b> (key, value)	number string		Puts raw byte data as a string value.
<b>putDouble</b> (key, value)	number number		Sets the value for a key whose type is double.
<b>putEnumerated</b> (key, enumType, value)	number number number		Sets the enumeration type and value for a key.

Method	Parameter type	Returns	What it does (Continued)
<b>putInteger</b> (key, value)	number number		Sets the value for a key whose type is integer.
<b>putLargeInteger</b> (key, value)	number number		Sets the value for a key whose type is large integer.
<b>putList</b> (key, value)	number <a href="#">ActionList</a>		Sets the value for a key whose type is an <code>ActionList</code> object.
<b>putObject</b> (key, classID, value)	number number <a href="#">ActionDescriptor</a>		Sets the value for a key whose type is an object, represented by an Action Descriptor.
<b>putPath</b> (key, value)	number <a href="#">File</a>		Sets the value for a key whose type is path.
<b>putReference</b> (key, value)	number <a href="#">ActionReference</a>		Sets the value for a key whose type is an object reference.
<b>putString</b> (key, value)	number string		Sets the value for a key whose type is string.
<b>putUnitDouble</b> (key, unitID, value)	number number number		Sets the value for a key whose type is a unit value formatted as a double.
<b>toStream</b> ( )		string	Gets the entire descriptor as a stream of bytes, for writing to disk.

# ActionList

This object provides an array-style mechanism for storing data. It can be used for low-level access into Photoshop.

This object is ideal when storing data of the same type. All items in the list must be of the same type.

You can use the "put" methods, such as [putBoolean\(\)](#), to append new elements, and can clear the entire list using [clear\(\)](#), but cannot otherwise modify the list.

**Note:** The `ActionList` object is part of the Action Manager functionality. For details on using the Action Manager, see the *Photoshop CC 2015 Scripting Guide*.

## Properties

Property	Value type	What it is
<b>count</b>	number	Read-only. The number of commands that comprise the action.
<b>typename</b>	string	Read-only. The class name of the referenced <code>ActionList</code> object.

## Methods

With the exception of the `clear()` method, you use the methods of this object to either get the value of a specific type of data in the list or set (put) the value type.

Method	Parameter type	Returns	What it does
<b>clear</b> ( )			Clears the list.
<b>getBoolean</b> (index)	number	boolean	Gets the value of a list element of type boolean.
<b>getClass</b> (index)	number	number	Gets the value of a list element of type class.
<b>getData</b> (index)	number	string	Gets raw byte data as a string value.
<b>getDouble</b> (index)	number	number	Gets the value of a list element of type double.
<b>getEnumerationType</b> (index)	number	number	Gets the enumeration type of a list element.
<b>getEnumerationValue</b> (index)	number	number	Gets the enumeration value of a list element.
<b>getInteger</b> (index)	number	number	Gets the value of a list element of type integer.



Method	Parameter type	Returns	What it does (Continued)
<b>getLargeInteger</b> (index)	number	number	Gets the value of a list element of type large integer.
<b>getList</b> (index)	number	<a href="#">ActionList</a>	Gets the value of a list element of type list.
<b>getObjectType</b> (index)	number	number	Gets the class ID of a list element of type object.
<b>getObjectValue</b> (index)	number	<a href="#">ActionDescriptor</a>	Gets the value of a list element of type object.
<b>getPath</b> (index)	number	<a href="#">File</a>	Gets the value of a list element of type File.
<b>getReference</b> (index)	number	<a href="#">ActionReference</a>	Gets the value of a list element of type <a href="#">ActionReference</a> .
<b>getString</b> (index)	number	string	Gets the value of a list element of type string.
<b>getType</b> (index)	number	<a href="#">DescValueType</a>	Gets the type of a list element.
<b>getUnitDoubleType</b> (index)	number	number	Gets the unit value type of a list element of type Double.
<b>getUnitDoubleValue</b> (index)	number	number	Gets the unit value of a list element of type double.
<b>putBoolean</b> (value)	boolean		Appends a new value, true or false.
<b>putClass</b> (value)	number		Appends a new value, a class or data type.
<b>putData</b> (value)	string		Appends a new value, a string containing raw byte data.
<b>putDouble</b> (value)	number		Appends a new value, a double.
<b>putEnumerated</b> (enumType, value)	number number		Appends a new value, an enumerated (constant) value.
<b>putInteger</b> (value)	number		Appends a new value, an integer.
<b>putLargeInteger</b> (value)	number		Appends a new value, a large integer.
<b>putList</b> (value)	<a href="#">ActionList</a>		Appends a new value, a nested action list.

Method	Parameter type	Returns	What it does (Continued)
<b>putObject</b> (classID, value)	number <a href="#">ActionDescriptor</a>		Appends a new value, an object.
<b>putPath</b> (value)	<a href="#">File</a>		Appends a new value, a path.
<b>putReference</b> (value)	<a href="#">ActionReference</a>		Appends a new value, a reference to an object created in the script.
<b>putString</b> (value)	string		Appends a new value, a string.
<b>putUnitDouble</b> (classID, value)	number number		Appends a new value, a unit/value pair.

# ActionReference

This object provides information about what the action is referring to. For example, when referring to the name of something you might use `keyName`. The reference would also need to know what name you are referring to. In this case you could use `classDocument` for the name of the document or `classLayer` for the name of the layer. It can be used for low-level access into `Photoshop.Contains` data associated with an [ActionDescriptor](#).

## Properties

Property	Value type	What it does
<code>typename</code>	string	Read-only. The class name of the referenced Action object.

## Methods

Method	Parameter type	Returns	What it does
<code>getContainer</code> ( )		<a href="#">ActionReference</a>	Gets a reference contained in this reference.  Container references provide additional pieces to the reference. This looks like another reference, but it is actually part of the same reference.
<code>getDesiredClass</code> ( )		number	Gets a number representing the class of the object.
<code>getEnumeratedType</code> ( )		number	Gets the enumeration type.
<code>getEnumeratedValue</code> ( )		number	Gets the enumeration value.
<code>getForm</code> ( )		<a href="#">ReferenceFormType</a>	Gets the form of this action reference.
<code>getIdentifier</code> ( )		number	Gets the identifier value for a reference whose form is identifier.
<code>getIndex</code> ( )		number	Gets the index value for a reference in a list or array.
<code>getName</code> ( )		string	Gets the name of a reference.
<code>getOffset</code> ( )		number	Gets the offset of the object's index value.
<code>getProperty</code> ( )		number	Gets the property ID value.

Method	Parameter type	Returns	What it does (Continued)
<b>putClass</b> (desiredClass)	number		Puts a new class form and class type into the reference.
<b>putEnumerated</b> (desiredClass, enumType, value)	number number number		Puts an enumeration type and ID into a reference along with the desired class for the reference.
<b>putIdentifier</b> (desiredClass, value)	number number		Puts a new identifier and value into the reference.
<b>putIndex</b> (desiredClass, value)	number number		Puts a new index and value into the reference.
<b>putName</b> (desiredClass, value)	number string		Puts a new name and value into the reference.
<b>putOffset</b> (desiredClass, value)	number number		Puts a new offset and value into the reference.
<b>putProperty</b> (desiredClass, value)	number number		Puts a new property and value into the reference.

# Application

The Adobe Adobe Photoshop CC 2015 application object, which is the root of the object model and provides access to all other objects. This object provides application-wide information, such as application defaults and available fonts. It provides many important methods, such as those for opening files and loading documents.

To access the properties and methods, you can use the pre-defined global variable `app`. For example:

```
var docRef = app.documents.add(800, 600, 72, "docRef", NewDocumentMode.RGB);
```

The properties and methods of the `Application` object are also available at the top level; you can omit references to the `Application` object altogether. For example:

```
var docRef = documents.add(800, 600, 72, "docRef", NewDocumentMode.RGB);
```

This usage can be somewhat ambiguous; for clarity, it is recommended that you use an explicit reference to `app`.

## Properties

Property	Value type	What it is
<b>activeDocument</b>	<a href="#">Document</a>	Read-write. The frontmost document.  Setting this property is equivalent to clicking an open document in the Adobe Photoshop CC 2015 application to bring it to the front of the screen.  <b>Tip:</b> If there is no open document, accessing this property throws an exception.
<b>backgroundColor</b>	<a href="#">SolidColor</a>	Read-write. The default background color and color style for documents.
<b>build</b>	string	Read-only. Information about the application.
<b>colorSettings</b>	string	Read-write. The name of the current color settings, as selected with Edit > Color Settings.
<b>currentTool</b>	string	Read-write. The name of the current tool selected.
<b>displayDialogs</b>	<a href="#">DialogModes</a>	Read-write. The dialog mode for the application, which controls what types of dialogs should be displayed when running scripts.
<b>documents</b>	<a href="#">Documents</a>	Read-only. The collection of open documents.  This is the primary point of access for documents that are currently open in the application. The array allows you to access any open document, or to iterate through all open documents.
<b>fonts</b>	<a href="#">TextFonts</a>	Read-only. The fonts installed on this system.
<b>foregroundColor</b>	<a href="#">SolidColor</a>	Read-write. The default foreground color (used to paint, fill, and stroke selections).

Property	Value type	What it is (Continued)
<b>freeMemory</b>	number	Read-only. The amount of unused memory available to Adobe Photoshop CC 2015.
<b>locale</b>	string	Read-only. The language location of the application.  An Adobe locale code consists of a 2-letter ISO-639 language code and an optional 2-letter ISO 3166 country code separated by an underscore. Case is significant. For example, en_US, en_UK, ja_JP, de_DE, fr_FR.
<b>macintoshFileTypes</b>	array of string	Read-only. A list of file image types Adobe Photoshop CC 2015 can open.
<b>measurementLog</b>	<a href="#">MeasurementLog</a>	The log of measurements taken.
<b>name</b>	string	Read-only. The application's name.
<b>notifiers</b>	<a href="#">Notifiers</a>	Read-only. The collection of notifiers currently configured (in the Scripts Events Manager menu in the Adobe Photoshop CC 2015 application).
<b>notifiersEnabled</b>	boolean	Read-write. True if all notifiers are enabled.
<b>path</b>	<a href="#">File</a>	Read-only. The full path to the location of the Adobe Photoshop CC 2015 application.
<b>playbackDisplayDialogs</b>	<a href="#">DialogModes</a>	Read-write. The dialog mode for playback mode, which controls what types of dialog to display when playing back a recorded action with the Actions palette.
<b>playbackParameters</b>	<a href="#">ActionDescriptor</a>	Read-write. Stores and retrieves parameters used as part of a recorded action. Can be used, for example, to control playback speed.
<b>preferences</b>	<a href="#">Preferences</a>	Read-only. The application preference settings (equivalent to selecting <b>Edit &gt; Preferences</b> in the Adobe Photoshop CC 2015 application in Windows or Photoshop > Preferences in Mac OS).
<b>preferencesFolder</b>	<a href="#">File</a>	Read-only. The full path to the Preferences folder.
<b>recentFiles</b>	array of <a href="#">File</a>	Read-only. Files in the Recent Files list.
<b>scriptingBuildDate</b>	string	Read-only. The build date of the Scripting interface.
<b>scriptingVersion</b>	string	Read-only. The version of the Scripting interface.
<b>systemInformation</b>	string	Read-only. Runtime details of the application and system.
<b>typename</b>	string	Read-only. The class name of the referenced app object.

Property	Value type	What it is (Continued)
<b>version</b>	string	Read-only. The version of Adobe Photoshop application you are running.
<b>windowsFileTypes</b>	array of string	Read-only. A list of file image extensions Adobe Photoshop CC 2015 can open.

## Methods

Method	Parameter type	Returns	What it does
<b>batch</b> (inputFiles, action, from [, options])	array of <a href="#">File</a> string string <a href="#">BatchOptions</a>	string	Runs the batch automation routine (similar to the <b>File &gt; Automate &gt; Batch</b> command).  The <code>inputFiles</code> parameter specifies the sources for the files to be manipulated by the batch command.
<b>beep</b> ( )			Causes a "beep" sound.
<b>bringToFront</b> ( )			Makes Adobe Photoshop CC 2015 the active (front-most) application.
<b>changeProgressText</b> (progressString)	string		Changes the text that appears in the progress window.  The <code>progressString</code> parameter is the string to show in the progress window.
<b>charIDToTypeID</b> (charID)	string	number	Converts from a four character code (character ID) to a runtime ID.
<b>doAction</b> (action, from)	string string		Plays an action from the Actions palette. The <code>action</code> parameter is the name of the action, the <code>from</code> parameter is the name of the action set.
<b>doForcedProgress</b> (progressString, javaScriptString)	string string		Performs a task with a progress bar. Forces progress bar to display, ignoring the normal heuristics that keep it from showing unnecessarily (for example, during very short tasks). Other progress APIs must be called periodically to update the progress bar and allow canceling.  The <code>progressString</code> parameter is the string to show in the progress window, the <code>javaScriptString</code> parameter is the string to execute.

Method	Parameter type	Returns	What it does (Continued)
<b>doProgress</b> (progressString, javaScriptString)	string string		<p>Performs a task with a progress bar. Other progress APIs must be called periodically to update the progress bar and allow canceling.</p> <p>The <code>progressString</code> parameter is the string to show in the progress window, the <code>javaScriptString</code> parameter is the string to execute.</p>
<b>doProgressSegmentTask</b> (segmentLength, done, total, javaScriptString)	number number number string	boolean	<p>Sections-off a portion of the unused progress bar for execution of a subtask. Returns false on cancel. This method should be used when iterating a list of tasks with unequal run times.</p> <p>The <code>segmentLength</code> parameter is the length of the current task, the <code>done</code> parameter is the total length of all completed tasks, the <code>total</code> parameter is the total length of all tasks. The <code>javaScriptString</code> parameter is the string to execute.</p>
<b>doProgressSubTask</b> (index, limit, javaScriptString)	number number string	boolean	<p>Sections-off a portion of the unused progress bar for execution of a subtask. Returns false on cancel. This method should be used when iterating a list of tasks with equal run times.</p> <p>The <code>index</code> parameter is the 0-based index of the current task, the <code>limit</code> parameter is the total number of tasks, and the <code>javaScriptString</code> parameter is the string to execute.</p>
<b>doProgressTask</b> (taskLength, javaScriptString)	number string	boolean	<p>Sections-off a portion of the unused progress bar for execution of a subtask. Returns false on cancel.</p> <p>The <code>taskLength</code> parameter is the amount of the unused progress bar to section-off between 0.0 and 1.0. The <code>javaScriptString</code> parameter is the string to execute.</p>
<b>eraseCustomOptions</b> (key)	string		Erases the user object with specified ID value from the Photoshop registry.



Method	Parameter type	Returns	What it does (Continued)
<b>executeAction</b> (eventID [, descriptor] [, displayDialogs])	number <a href="#">ActionDescriptor</a> <a href="#">DialogModes</a>	<a href="#">ActionDescriptor</a>	Plays an Action Manager event.
<b>executeActionGet</b> (reference)	<a href="#">ActionReference</a>	<a href="#">ActionDescriptor</a>	Obtains information about a predefined or recorded action.
<b>featureEnabled</b> (name)	string	boolean	Determines whether the feature specified by name is enabled.  The following features are supported as values for name:  "photoshop/extended" "photoshop/standard" "photoshop/trial"
<b>getCustomOptions</b> (key)	string	<a href="#">ActionDescriptor</a>	Retrieves user objects in the Photoshop registry for the ID with value key.
<b>isQuicktimeAvailable</b> ( )		boolean	Returns true if Quicktime is installed.
<b>load</b> (document)	<a href="#">File</a>		Loads a support file (as opposed to a Photoshop image document) from the specified location.
<b>makeContactSheet</b> (inputFiles [, options])	array of <a href="#">File</a> <a href="#">ContactSheetOptions</a>	string	DEPRECATED for Adobe Photoshop CS4.
<b>makePDFPresentation</b> (inputFiles, outputFiles [, options])	array of <a href="#">File</a> <a href="#">File</a> <a href="#">PresentationOptions</a>	string	DEPRECATED for Adobe Photoshop CS4.
<b>makePhotoGallery</b> (inputFolder, outputFolder [, options])	<a href="#">File</a> <a href="#">File</a> <a href="#">GalleryOptions</a>	string	DEPRECATED for Adobe Photoshop CS4.
<b>makePhotomerge</b> (inputFiles)	array of <a href="#">File</a>	string	DEPRECATED for Adobe Photoshop CC 2015. Use provided script:  runphotomergeFromScript = true; \$.evalFile( app.path + "Presets/Scripts/Photomerge.jsx") photomerge.createPanorama( fileList, displayDialog );  Merges multiple files into one, with user interaction required.
<b>makePicturePackage</b> (inputFiles [, options])	array of <a href="#">File</a> <a href="#">PicturePackageOptions</a>	string	DEPRECATED for Adobe Photoshop CS4.

Method	Parameter type	Returns	What it does (Continued)
<b>open</b> (document [, as] [, asSmartObject])	<a href="#">File</a> object or <a href="#">OpenDocumentType</a> boolean	<a href="#">Document</a>	<p>Opens the specified document.</p> <p>Use the optional <code>as</code> parameter to specify the file format using the constants in <a href="#">OpenDocumentType</a>; or, you can specify a file format together with its open options using these objects:</p> <p> <a href="#">CameraRAWOpenOptions</a>  <a href="#">DICOMOpenOptions</a>  <a href="#">EPSOpenOptions</a>  <a href="#">PDFOpenOptions</a>  <a href="#">PhotoCDOpenOptions</a>  <a href="#">RawFormatOpenOptions</a> </p> <p>Use the optional parameter <code>asSmartObject</code> (default: <code>false</code>) to create a smart object around the opened document.</p> <p>See the <a href="#">Application sample scripts</a> for an example of using the <a href="#">File</a> object in the <code>open</code> method.</p>
<b>openDialog</b> ()		array of <a href="#">File</a>	<p>Invokes the Photoshop Open dialog box for the user to select files.</p> <p>Returns an array of <a href="#">File</a> objects for the files selected in the dialog.</p>
<b>purge</b> (target)	<a href="#">PurgeTarget</a>		Purges one or more caches.
<b>putCustomOptions</b> (key, customObject [, persistent])	string <a href="#">ActionDescriptor</a> boolean		<p>Saves a customized settings object in the Photoshop registry.</p> <p><code>key</code> is the unique identifier for your custom settings.</p> <p><code>customObject</code> is the object to save in the registry.</p> <p><code>persistent</code> indicates whether the object should persist once the script has finished.</p>
<b>refresh</b> ()			<p>Pauses the script while the application refreshes. Use to slow down execution and show the results to the user as the script runs. Use carefully; your script runs much more slowly when using this method.</p>
<b>refreshFonts</b> ()			Force the font list to get updated.
<b>runMenuItem</b> (menuID)	number		Run a menu item given the menu ID.

Method	Parameter type	Returns	What it does (Continued)
<b>showColorPicker</b> ( )		boolean	Returns false if dialog is cancelled, true otherwise.
<b>stringIDToTypeID</b> (stringID)	string	number	Converts from a string ID to a runtime ID.
<b>togglePalettes</b> ( )			Toggle palette visibility.
<b>toolSupportsBrushes</b> (tool)	string	boolean	Returns true if the specified tool supports brushes, false otherwise.
<b>TypeIDToCharID</b> (TypeID)	number	string	Converts from a runtime ID to a character ID.
<b>TypeIDToStringID</b> (TypeID)	number	string	Converts from a runtime ID to a string ID.
<b>updateProgress</b> (done, total)	number number	boolean	<p>Updates the progress bar started by <b>doProgress</b> method. This method should be used for manual non-task based progress updating. Returns false on cancel.</p> <p>The <b>done</b> parameter is the number of tasks completed and the <b>total</b> parameter the total number of tasks.</p>

## Application sample scripts

### Application.jsx

This script invokes an alert box to display Properties important to an application such as version number, the path to the application, the amount of memory available, and the number of documents open.

When a user presses the OK button on the alert box, a second dialog opens, which asks users whether they would like the foreground and background colors set for the document presently open. If no document is open, the script opens a new document for the user.

The script (with no document open) produces a progression of three dialogs.

```
//Create a Welcome message
// Use the name and version properties of the application object to
// Append the application's name and version to the Welcome message
// use "\r" to insert a carriage return
// use the combination operator += to append info to the message
var message = "Welcome to " + app.name
message += " version " + app.version + "\r\r"

// find out where Adobe Photoshop CC 2015 is installed
// and add the path to the message
// add the optional parameter fsName to the path property
// to display the file system name in the most common format
message += "I'm installed in " + app.path.fsName + "\r\r"

// see how much memory Adobe Photoshop CC 2015 has to play with
```

```
message += "You have this much memory available for Adobe Photoshop CC 2015: " +
app.freeMemory + "\r\r"

// use the length property of the documents object to
// see how many documents are open
var documentsOpen = app.documents.length
message += "You currently have " + documentsOpen + " document(s) open.\r\r"

// display the message to the user
alert(message)

// answer will be true for a "Yes" answer and false for a "No" answer
var answer = confirm("Set the foreground and background to my favorite colors?")

// set the colors
if (answer) {
    // I don't have a favorite color. Why did I ask you may wonder?
    app.foregroundColor.rgb.red = Math.random() * 255
    app.foregroundColor.rgb.green = Math.random() * 255
    app.foregroundColor.rgb.blue = Math.random() * 255
    app.backgroundColor.rgb.red = Math.random() * 255
    app.backgroundColor.rgb.green = Math.random() * 255
    app.backgroundColor.rgb.blue = Math.random() * 255
}

// Open a document
if (app.documents.length == 0) {

    // use the application's path and the offset to the samples folder
    var sampleDocToOpen = File(app.path + "/Samples/Fish.psd")

    // compose a message with the name of the file
    message = "Would you like me to open a sample for you? ("
    message += sampleDocToOpen.fileName
    message += ")"

    // ask the user another question
    answer = confirm(message)

    // open the document accordingly
    if (answer) {
        open(sampleDocToOpen)
    }
}
```

### PDFPresentation.jsx

This script presents a progression of images as an Adobe PDF slide show.

```
// use all the files in the Samples folder
var inputFolder = new Folder(app.path + "/Samples/")

// see if we have something interesting
if (inputFolder != null) {

    // get all the files found in this folder that are Adobe Photoshop CC 2015
    (.psd format)
    var inputFiles = inputFolder.GetFiles("*.psd")

    // output to the desktop
```

```
var outputFile = File("~/Desktop/JavaScriptPresentation.pdf")

// there are defaults but I like to set the options myself
var options = new PresentationOptions
options.presentation = true
options.view = true
options.autoAdvance = true
options.interval = 5
options.loop = true
options.transition = TransitionType.RANDOM

// create the presentation
makePDFPresentation(inputFiles, outputFile, options)

alert("Presentation file saved to: " + outputFile.fsName)
}
```

# ArtLayer

An object within a document that contains the visual elements of the image (equivalent to a layer in the Adobe Photoshop CC 2015 application).

Access an art layer in a document through the [Document.artLayers](#) collection. You can access a layer by name; for example:

```
var layerRef = app.activeDocument.artLayers.getByName("my layer");  
layerRef.allLocked = true;
```

Access the art layers in a layer set through the [LayerSet.artLayers](#) collection in the parent set.

## Properties

Property	Value type	What it is
<b>allLocked</b>	boolean	Read-write. True to completely lock the contents and settings of this layer.
<b>blendMode</b>	<a href="#">BlendMode</a>	Read-write. The blending mode.
<b>bounds</b>	array of <a href="#">UnitValue</a>	Read-only. An array of coordinates that describes the bounding rectangle of the layer.
<b>boundsNoEffects</b>	array of <a href="#">UnitValue</a>	Read-only. An array of coordinates that describes the bounding rectangle of the layer not including effects.
<b>fillOpacity</b>	number [0.0..100]	Read-write. The interior opacity of the layer, a percentage value.
<b>filterMaskDensity</b>	double	Read-write. The density of the filter mask (between 0.0 and 250.0)
<b>filterMaskFeather</b>	double	Read-write. The feather of the filter mask (between 0.0 and 250.0)
<b>grouped</b>	boolean	Read-write. True if this layer is grouped with the layer beneath it.
<b>isBackgroundLayer</b>	boolean	Read-write. True if this is the background layer of the document. A document can have only one background layer. If there is no background layer, setting this to true causes this to become the background layer.
<b>kind</b>	<a href="#">LayerKind</a>	<p>Read-write. Sets the type (such as 'text layer') for an empty layer.</p> <p>Valid only when the layer is empty and when <code>isBackgroundLayer</code> is false. See <a href="#">isBackgroundLayer</a>.</p> <p>You can use the <code>kind</code> property to make a background layer a normal layer; however, to make a layer a background layer, you must set <code>isBackgroundLayer</code> to true.</p>

Property	Value type	What it is (Continued)
<b>layerMaskDensity</b>	double	Read-write. The density of the layer mask (between 0.0 and 100.0)
<b>layerMaskFeather</b>	double	Read-write. The feather of the layer mask (between 0.0 and 250.0)
<b>linkedLayers</b>	array of <a href="#">ArtLayer</a> or <a href="#">LayerSet</a>	Read-only. The layers linked to this layer. See <a href="#">ArtLayer.link</a> .
<b>name</b>	string	Read-write. The name.
<b>opacity</b>	number [0.0..100.0].	Read-write. The master opacity of the layer, a percentage value.
<b>parent</b>	<a href="#">Document</a>	Read-only. The object's container.
<b>pixelsLocked</b>	boolean	Read-write. True if the pixels in the layer's image cannot be edited using the paintbrush tool.
<b>positionLocked</b>	boolean	Read-write. True if the pixels in the layer's image cannot be moved within the layer.
<b>textItem</b>	<a href="#">TextItem</a>	Read-only. The text item that is associated with the layer.  Valid only when <a href="#">kind</a> = <code>LayerKind.TEXT</code> .
<b>transparentPixelsLocked</b>	boolean	Read-write. True if editing is confined to the opaque portions of the layer.
<b>typename</b>	string	Read-only. The class name of the referenced <code>artLayer</code> object.
<b>vectorMaskDensity</b>	double	Read-write. The density of the vector mask (between 0.0 and 250.0)
<b>vectorMaskFeather</b>	double	Read-write. The feather of the vector mask (between 0.0 and 250.0)
<b>visible</b>	boolean	Read-write. True if the layer is visible.
<b>xmpMetadata</b>	<a href="#">xmpMetadata</a>	Read-write. Metadata for the layer.

## Methods

Method	Parameter type	Returns	What it does
<b>adjustBrightnessContrast</b> (brightness, contrast)	number number		Adjusts the brightness in the range [-100..100] and contrast [-100..100].
<b>adjustColorBalance</b> ([shadows] [, midtones] [, highlights] [, preserveLuminosity])	array of number array of number array of number boolean		Adjusts the color balance of the layer's component channels. For shadows, midtones, and highlights, the array must include three values in the range [-100..100], which represent cyan or red, magenta or green, and yellow or blue, when the document mode is CMYK or RGB.  See <a href="#">Document.mode</a> .
<b>adjustCurves</b> (curveShape)	array of array of number		Adjusts the tonal range of the selected channel using up to fourteen points.  Each value in the <code>curveShape</code> array is a point pair, an array of an x and y integer value.
<b>adjustLevels</b> (inputRangeStart, inputRangeEnd, inputRangeGamma, outputRangeStart, outputRangeEnd)	number [0..253] number [(start + 2)..255] number [0.10..9.99] number [0..253] number [(start + 2)..255]		Adjusts the levels of the selected channels
<b>applyAddNoise</b> (amount, distribution, monochromatic)	number [0.1..400] <a href="#">NoiseDistribution</a> boolean		Applies the Add Noise filter amount is a percentage value.
<b>applyAverage</b> ( )			Applies the Average filter.
<b>applyBlur</b> ( )			Applies the Blur filter.
<b>applyBlurMore</b> ( )			Applies the Blur More filter.
<b>applyClouds</b> ( )			Applies the Clouds filter.
<b>applyCustomFilter</b> (characteristics, scale, offset)	array of number number number		Applies a custom filter. The <code>characteristics</code> array has 25 members.  See Adobe Photoshop CC 2015 Help for specific instructions.



Method	Parameter type	Returns	What it does (Continued)
<b>applyDeInterlace</b> (eliminateFields, createFields)	<a href="#">EliminateFields</a> <a href="#">CreateFields</a>		Applies the De-Interlace filter.
<b>applyDespeckle</b> ( )			Applies the Despeckle filter.
<b>applyDifferenceClouds</b> ( )			Applies the Difference Clouds filter.
<b>applyDiffuseGlow</b> (graininess, glowAmount, clearAmount)	number [0..10] number [0..20] number [0..20]		Applies the Diffuse Glow filter.
<b>applyDisplace</b> (horizontalScale, verticalScale, displacement, undefinedareas, displacementMapFiles)	number [-999..999] number [-999..999] <a href="#">DisplacementMapType</a> <a href="#">UndefinedAreas</a> <a href="#">File</a>		Applies the Displace filter using the specified horizontal and vertical scale, mapping type, treatment of undistorted areas, and path to the distortion image map.
<b>applyDustAndScratches</b> (radius, threshold)	number [1..100] number [0..255]		Applies the Dust & Scratches filter.
<b>applyGaussianBlur</b> (radius)	number [0.1..250.0]		Applies the Gaussian Blur filter within the specified radius (in pixels)
<b>applyGlassEffect</b> (distortion, smoothness, scaling [, invert] [, texture] [, textureFile])	number [0..20] number [1..15] number [50..200] boolean <a href="#">TextureType</a> <a href="#">File</a>		Applies the Glass filter. scaling is a percentage value.
<b>applyHighPass</b> (radius)	number [0.1..250.0]		Applies the High Pass filter within the specified radius.

Method	Parameter type	Returns	What it does (Continued)
<b>applyLensBlur</b> ([source] [, focalDistance] [, invertDepthMap] [, shape] [, radius] [, bladeCurvature] [, rotation] [, brightness] [, threshold] [, amount] [, distribution] [, monochromatic] )	<a href="#">DepthMapSource</a> number boolean <a href="#">Geometry</a> number number number number <a href="#">NoiseDistribution</a> boolean		<p>Applies the Lens Blur filter.</p> <p><b>source:</b> The source for the depth map (default: <code>DepthMapSource.NONE</code>)</p> <p><b>focalDistance:</b> The blur focal distance for the depth map (default: 0).</p> <p><b>invertDepthMask:</b> True if the depth map is inverted (default: false).</p> <p><b>shape:</b> The shape of the iris (default: <code>Geometry.HEXAGON</code>)</p> <p><b>radius:</b> The radius of the iris (default: 15).</p> <p><b>bladeCurvature:</b> The blade curvature of the iris (default: 0).</p> <p><b>rotation:</b> The rotation of the iris (default: 0)</p> <p><b>brightness:</b> The brightness for the specular highlights (default: 0).</p> <p><b>threshold:</b> The threshold for the specular highlights (default: 0).</p> <p><b>amount:</b> The amount of noise (default: 0)</p> <p><b>distribution:</b> The distribution value for the noise (default: <code>NoiseDistribution.UNIFORM</code>).</p> <p><b>monochromatic:</b> True if the noise is monochromatic (default: false).</p>
<b>applyLensFlare</b> (brightness, flareCenter, lensType)	number array ( <a href="#">UnitValue</a> ) <a href="#">LensType</a>		<p>Applies the Lens Flare filter with the specified brightness (0 - 300, as a percentage), the x and y coordinates (unit value) of the flare center, and the lens type.</p>
<b>applyMaximum</b> (radius)	number [1..100]		<p>Applies the Maximum filter within the specified radius (in pixels).</p>
<b>applyMedianNoise</b> (radius)	number [1..100]		<p>Applies the Median Noise filter within the specified radius (in pixels).</p>

Method	Parameter type	Returns	What it does (Continued)
<b>applyMinimum</b> (radius)	number [1..100]		Applies the Minimum filter within the specified radius (in pixels) (1 - 100).
<b>applyMotionBlur</b> (angle, radius)	number [-360..360] number [1..999]		Applies the Motion Blur filter.
<b>applyNTSC</b> ( )			Applies the NTSC colors filter.
<b>applyOceanRipple</b> (size, magnitude)	number [1..15] number [0..20]		Applies the Ocean Ripple filter.
<b>applyOffset</b> (horizontal, vertical, undefinedAreas)	<a href="#">UnitValue</a> <a href="#">UnitValue</a> <a href="#">OffsetUndefinedAreas</a>		Moves the layer the specified amount horizontally and vertically (min/max amounts depend on layer size), leaving an undefined area at the layer's original location.
<b>applyPinch</b> (amount)	number [-100..100]		Applies the Pinch filter. amount is a percentage value.
<b>applyPolarCoordinates</b> (conversion)	<a href="#">PolarConversionType</a>		Applies the Polar Coordinates filter.
<b>applyRadialBlur</b> (amount, blurMethod, blurQuality [, blurCenter] )	number [1..100] <a href="#">RadialBlurMethod</a> <a href="#">RadialBlurQuality</a> <a href="#">UnitValue</a>		Applies the Radial Blur filter in the specified amount, using either a spin or zoom effect and the specified quality. The parameter <code>blurCenter</code> is the position (unit value).
<b>applyRipple</b> (amount, size)	number [-999..999] <a href="#">RippleSize</a>		Applies the Ripple filter in the specified amount, throughout the image and in the specified size.
<b>applySharpen</b> ( )			Applies the Sharpen filter.
<b>applySharpenEdges</b> ( )			Applies the Sharpen Edges filter.
<b>applySharpenMore</b> ( )			Applies the Sharpen More filter.

Method	Parameter type	Returns	What it does (Continued)
<b>applyShear</b> (curve, undefinedAreas)	array of array of number <a href="#">UndefinedAreas</a>		Applies the Shear filter. The <code>curve</code> defines a curve with [2..255] points.  Each value in the <code>curve</code> array is a point pair, an array of an x and y integer value.
<b>applySmartBlur</b> (radius, threshold, blurQuality, mode)	number [0.1..100.0] number [0.1..100.0] <a href="#">SmartBlurQuality</a> <a href="#">SmartBlurMode</a>		Applies the Smart Blur filter.
<b>applySpherize</b> (amount, mode)	number [-100..100] <a href="#">SpherizeMode</a>		Applies the Spherize filter.  <code>amount</code> is a percentage value.
<b>applyStyle</b> (styleName)	string		Applies the specified style to the layer. You must use a style from the Styles list in the Layer Styles Palette.
<b>applyTextureFill</b> (textureFile)	<a href="#">File</a>		Applies the Texture Fill filter.
<b>applyTwirl</b> (angle)	number [-999..999]		Applies the Twirl filter.
<b>applyUnsharpMask</b> (amount, radius, threshold)	number [1..500] number [0.1..250.0] number [0..255]		Applies the Unsharp Mask filter. ( <code>amount</code> is a percentage value.)
<b>applyWave</b> (generatorNumber, minimumWavelength, maximumWavelength, minimumAmplitude, maximumAmplitude, horizontalScale, verticalScale, waveType, undefinedAreas, randomSeed)	number [1..999] number [1..998] number [2.. <i>min</i> +1] number [1..998] number [2.. <i>min</i> +1] number [1..100] number [1..100] <a href="#">WaveType</a> <a href="#">UndefinedAreas</a> number		Applies the Wave filter.  Scale factors are percentage values.
<b>applyZigZag</b> (amount, ridges, style)	number [-100..100] number [0..20] <a href="#">ZigZagType</a>		Applies the Zigzag filter.
<b>autoContrast</b> ( )			Adjusts the contrast of the selected channels automatically.
<b>autoLevels</b> ( )			Adjusts the levels of the selected channels using the auto levels option.

Method	Parameter type	Returns	What it does (Continued)
<b>clear</b> ( )			Cuts the layer without moving it to the clipboard.
<b>copy</b> ( [merge] )	boolean		Copies the layer to the clipboard. When the optional argument is set to <code>true</code> , a merged copy is performed (that is, all visible layers are copied to the clipboard).
<b>cut</b> ( )			Cuts the layer to the clipboard.
<b>desaturate</b> ( )			Converts a color image to a grayscale image in the current color mode by assigning equal values of each component color to each pixel.
<b>duplicate</b> ( [relativeObject] [, insertionLocation] )	<a href="#">ArtLayer</a> or <a href="#">LayerSet</a> <a href="#">ElementPlacement</a>	<a href="#">ArtLayer</a> or <a href="#">LayerSet</a>	Creates a duplicate of the object on the screen.
<b>equalize</b> ( )			Redistributes the brightness values of pixels in an image to more evenly represent the entire range of brightness levels within the image.
<b>invert</b> ( )			Inverts the colors in the layer by converting the brightness value of each pixel in the channels to the inverse value on the 256-step color-values scale.
<b>link</b> (with)	<a href="#">ArtLayer</a> or <a href="#">LayerSet</a>		Links the layer with the specified layer.
<b>merge</b> ( )		<a href="#">ArtLayer</a>	Merges the layer down, removing the layer from the document; returns a reference to the art layer that this layer is merged into.

Method	Parameter type	Returns	What it does (Continued)
<b>mixChannels</b> (outputChannels [, monochrome])	array of array of number boolean		<p>Modifies a targeted (output) color channel using a mix of the existing color channels in the image.</p> <p>The <code>outputChannels</code> parameter is an array of channel specifications. For each component channel, specify a list of adjustment values in the range [-200..200] followed by a 'constant' value [-200..200].)</p> <p>When <code>monochrome = true</code>, the maximum number of channel value specifications is 1.</p> <p>Valid only when <code>docRef.mode = DocumentMode.RGB</code> or <code>DocumentMode.CMYK</code>.</p> <p>RGB arrays must include four values. CMYK arrays must include five values.</p>
<b>move</b> (relativeObject, insertionLocation)	<a href="#">ArtLayer</a> or <a href="#">LayerSet</a> <a href="#">ElementPlacement</a>		<p>Moves the layer relative to the object specified in parameters.</p> <p>For art layers, only the constant values <code>ElementPlacement.PLACEBEFORE</code> and <code>ElementPlacement.PLACEAFTER</code> are valid.</p> <p>For layer sets, only the constant values <code>ElementPlacement.PLACEBEFORE</code> and <code>ElementPlacement.INSIDE</code> are valid.</p>
<b>photoFilter</b> ([fillColor] [, density] [, preserveLuminosity])	<a href="#">SolidColor</a> number [1..100] boolean		<p>Adjust the layer's color balance and temperature as if a color filter had been applied.</p> <p><code>density</code> is a percentage value.</p>
<b>posterize</b> (levels)	number [2..225]		<p>Specifies the number of tonal levels for each channel and then maps pixels to the closest matching level.</p>
<b>rasterize</b> (target)	<a href="#">RasterizeType</a>		<p>Converts the targeted contents in the layer into a flat, raster image.</p>
<b>remove</b> ()			<p>Deletes the object.</p>

Method	Parameter type	Returns	What it does (Continued)
<b>resize</b> ([horizontal] [, vertical] [, anchor])	number number <a href="#">AnchorPosition</a>		Resizes the layer to the specified dimensions (as a percentage of its current size) and places it in the specified position.
<b>rotate</b> (angle [, anchor])	number <a href="#">AnchorPosition</a>		Rotates rotates the layer around the specified anchor point (default: MIDDLECENTER).
<b>selectiveColor</b> (selectionMethod [, reds] [, yellows] [, greens] [, cyans] [, blues] [, magentas] [, whites] [, neutrals] [, blacks])	<a href="#">AdjustmentReference</a> array of number array of number array of number array of number array of number array of number array of number array of number		Modifies the amount of a process color in a specified primary color without affecting the other primary colors.  Each color array must have four values.
<b>shadowHighlight</b> ([shadowAmount] [, shadowWidth] [, shadowRadius] [, highlightAmount] [, highlightWidth] [, highlightRadius] [, colorCorrection] [, midtoneContrast] [, blackClip] [, whiteClip])	number [0..100] number [0..100] number [0..2500] number [0..100] number [0..100] number [0..2500] number [-100..100] number [-100..100] number [0.000..50.000] number [0.000..50.000]		Adjusts the range of tones in the image's shadows and highlights.  Amounts and widths are percentage values. Radius values are in pixels.
<b>threshold</b> (level)	number [1..255]		Converts grayscale or color images to high-contrast, B/W images by converting pixels lighter than the specified threshold to white and pixels darker than the threshold to black.
<b>translate</b> ([deltaX] [, deltaY])	<a href="#">UnitValue</a> <a href="#">UnitValue</a>		Moves the layer the specified amount (in the given unit) relative to its current position.
<b>unlink</b> ( )			Unlinks the layer.

## ArtLayer sample Script

The following script opens all the files in the samples folder, creating one multi-layered document. Each layer is pasted into one of four quadrants and given 50% transparency. Finally the layers are sorted by name.

### ArtLayer.jsx

```
// Save the current preferences
var startRulerUnits = app.preferences.rulerUnits
var startTypeUnits = app.preferences.typeUnits
var startDisplayDialogs = app.displayDialogs

// Set Adobe Photoshop CC 2015 to use pixels and display no dialogs
app.preferences.rulerUnits = Units.PIXELS
app.preferences.typeUnits = TypeUnits.PIXELS
app.displayDialogs = DialogModes.NO

//Close all the open documents
while (app.documents.length) {
    app.activeDocument.close()
}

// Create a new document to merge all the samples into
var mergedDoc = app.documents.add(1000, 1000, 72, "Merged Samples",
NewDocumentMode.RGB, DocumentFill.TRANSPARENT, 1)

// Use the path to the application and append the samples folder
var samplesFolder = Folder(app.path + "/Samples/")

//Get all the files in the folder
var fileList = samplesFolder.GetFiles()

// open each file
for (var i = 0; i < fileList.length; i++) {
    // The fileList is folders and files so open only files
    if (fileList[i] instanceof File) {
        open(fileList[i])

        // use the document name for the layer name in the merged document
        var docName = app.activeDocument.name

        // flatten the document so we get everything and then copy
        app.activeDocument.flatten()
        app.activeDocument.selection.selectAll()
        app.activeDocument.selection.copy()

        // don't save anything we did
        app.activeDocument.close(SaveOptions.DONOTSAVECHANGES)

        // make a random selection on the document to paste into
        // by dividing the document up in 4 quadrants and pasting
        // into one of them by selecting that area
        var topLeftH = Math.floor(Math.random() * 2)
        var topLeftV = Math.floor(Math.random() * 2)
        var docH = app.activeDocument.width.value / 2
        var docV = app.activeDocument.height.value / 2
        var selRegion = Array(Array(topLeftH * docH, topLeftV * docV),
            Array(topLeftH * docH + docH, topLeftV * docV),
            Array(topLeftH * docH + docH, topLeftV * docV + docV),
            Array(topLeftH * docH, topLeftV * docV + docV),
            Array(topLeftH * docH, topLeftV * docV))
        app.activeDocument.selection.select(selRegion)
        app.activeDocument.paste()

        // change the layer name and opacity
        app.activeDocument.activeLayer.name = docName
        app.activeDocument.activeLayer.fillOpacity = 50
    }
}
```



```
    }  
  }  
  
  // sort the layers by name  
  for (var x = 0; x < app.activeDocument.layers.length; x++) {  
    for (var y = 0; y < app.activeDocument.layers.length - 1 - x; y++) {  
      // Compare in a non-case sensitive way  
      var doc1 = app.activeDocument.layers[y].name  
      var doc2 = app.activeDocument.layers[y + 1].name  
      if (doc1.toUpperCase() > doc2.toUpperCase()) {  
        app.activeDocument.layers[y].move(app.activeDocument.layers[y+1],  
          ElementPlacement.PLACEAFTER)  
      }  
    }  
  }  
}  
  
// Reset the application preferences  
app.preferences.rulerUnits = startRulerUnits  
app.preferences.typeUnits = startTypeUnits  
app.displayDialogs = startDisplayDialogs
```

# ArtLayers

The collection of [ArtLayer](#) objects in a document or layer set.

Access through the [Document.artLayers](#) or [LayerSet.artLayers](#) collection. For example:

```
var layerRef = docRef.artLayers.add()
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>artLayers</code> collection.
<b>parent</b>	<a href="#">Document</a>	Read-only. The object's container.
<b>typename</b>	string	Read-only. The class name of the referenced <code>artLayers</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>add</b> ( )		<a href="#">ArtLayer</a>	Creates a new art layer in the document and adds the new object to this collection.
<b>getByName</b> (name)	string	<a href="#">ArtLayer</a>	Get the first element in the <code>artLayers</code> collection with the provided name.
<b>removeAll</b> ( )			Removes all elements from the <code>artLayers</code> collection.

## BatchOptions

Options for running a batch operation using the [Application.batch\(\)](#) method.

JavaScript only supports folders as sources for batch commands. Specify the batch source folder as the `inputFiles` parameter of the [Application.batch\(\)](#) method.

### Properties

Property	Value type	What it is
<b>destination</b>	<a href="#">BatchDestinationType</a>	Read-write. The type of destination for the processed files (default: <code>BatchDestinationType.NODESTINATION</code> ).
<b>destinationFolder</b>	<a href="#">Folder</a>	Read-write. The folder location for the processed files.  Valid only when <a href="#">destination</a> = <code>BatchDestinationType.FOLDER</code> .
<b>errorFile</b>	<a href="#">File</a>	Read-write. The file in which to log errors encountered.  To display errors on the screen (and stop batch processing when errors occur) leave blank.
<b>fileNaming</b>	array of <a href="#">FileNamingType</a>	Read-write. A list of file naming options (maximum: 6).  Valid only when <a href="#">destination</a> = <code>BatchDestinationType.FOLDER</code> .
<b>macintoshCompatible</b>	boolean	Read-write. True to make the final file names Macintosh compatible (default: <code>true</code> ).  Valid only when <a href="#">destination</a> = <code>BatchDestinationType.FOLDER</code> .
<b>overrideOpen</b>	boolean	Read-write. True to override action open commands (default: <code>false</code> ).
<b>overrideSave</b>	boolean	Read-write. True to override save as action steps with the specified destination (default: <code>false</code> ).  Valid only when <a href="#">destination</a> = <code>BatchDestinationType.FOLDER</code> or <code>SAVEANDCLOSE</code> .
<b>startingSerial</b>	number	Read-write. The starting serial number to use in naming files (default: 1).  Valid only when <a href="#">destination</a> = <code>BatchDestinationType.FOLDER</code> .
<b>suppressOpen</b>	boolean	Read-write. True to suppress the file open options dialogs (default: <code>false</code> ).

Property	Value type	What it is (Continued)
<b>suppressProfile</b>	boolean	Read-write. True to suppress the color profile warnings (default: <code>false</code> ).
<b>typename</b>	string	Read-only. The class name of the referenced <code>batchOptions</code> object.
<b>unixCompatible</b>	boolean	Read-write. True to make the final file name Unix compatible (default: <code>true</code> ).  Valid only when <a href="#">destination</a> = <code>BatchDestinationType.FOLDER</code> .
<b>windowsCompatible</b>	boolean	Read-write. True to make the final file names Windows compatible (default: <code>true</code> ).  Valid only when <a href="#">destination</a> = <code>BatchDestinationType.FOLDER</code> .

# BitmapConversionOptions

Options for converting an image to bitmap mode, using [Document.changeMode\(\)](#) with [ChangeMode.Bitmap](#).

Convert color images to grayscale before converting the image to bitmap mode. See the [ArtLayer.desaturate\(\)](#) method.

## Properties

Property	Value type	What it is
<b>angle</b>	number [-180..180]	Read-write. The angle (in degrees) at which to orient individual dots. See <a href="#">shape</a> .  Valid only when <a href="#">method</a> = <code>BitmapConversionType.HALFTONESCREEN</code> .
<b>frequency</b>	number [1.0..999.99]	Read-write. The number of printer dots (per inch) to use.  Valid only when <a href="#">method</a> = <code>BitmapConversionType.HALFTONESCREEN</code> .
<b>method</b>	<a href="#">BitmapConversionType</a>	Read-write. The conversion method to use (default: <code>BitmapConversionType.DIFFUSIONDITHER</code> ).
<b>patternName</b>	string	Read-write. The name of the pattern to use.  For information about pre-installed valid patterns, see Adobe Photoshop CC 2015 Help on the bitmap conversion command, or view the options available in the Custom Color drop down box after choosing the bitmap conversion command.  Valid only when <a href="#">method</a> = <code>BitmapConversionType.CUSTOMPATTERN</code> .
<b>resolution</b>	number	Read-write. The output resolution in pixels per inch (default: 72.0).
<b>shape</b>	<a href="#">BitmapHalfToneType</a>	Read-write. The dot shape to use.  Valid only when <a href="#">method</a> = <code>BitmapConversionType.HALFTONESCREEN</code> .
<b>typename</b>	string	Read-only. The class name of the referenced <code>bitmapConversionOptions</code> object.

## BMPSaveOptions

Options for saving a document in BMP format using the [Document.saveAs\(\)](#) method.

### Properties

Property	Value type	What it is
<b>alphaChannels</b>	boolean	Read-write. True to save the alpha channels.
<b>depth</b>	<a href="#">BMPDepthType</a>	Read-write. The number of bits per channel.
<b>flipRowOrder</b>	boolean	Read-write. True to write the image from top to bottom (default: <code>false</code> ).  Available only when <a href="#">osType</a> = <code>OperatingSystem.WINDOWS</code> .
<b>osType</b>	<a href="#">OperatingSystem</a>	Read-write. The target OS. (default: <code>OperatingSystem.WINDOWS</code> ).
<b>rleCompression</b>	boolean	Read-write. True to use RLE compression.  Available only when <a href="#">osType</a> = <code>OperatingSystem.WINDOWS</code> .
<b>typename</b>	string	Read-only. The class name of the referenced <code>BMPSaveOptions</code> object.

## CameraRAWOpenOptions

Options for opening a document in Camera RAW format using the [Application.open\(\)](#) method.

### Properties

Property	Value type	What it is
<b>bitsPerChannel</b>	<a href="#">BitsPerChannelType</a>	Read-write. The number of bits per channel.
<b>blueHue</b>	number [-100..100]	Read-write. The blue hue of the shot.
<b>blueSaturation</b>	number [-100..100]	Read-write. The blue saturation of the shot.
<b>brightness</b>	number [0..150]	Read-write. The brightness of the shot.
<b>chromaticAberrationBY</b>	number [-100..100]	Read-write. The chromatic aberration B/Y of the shot.
<b>chromaticAberrationRC</b>	number [-100..100]	Read-write. The chromatic aberration R/C of the shot.
<b>colorNoiseReduction</b>	number [0..100]	Read-write. The color noise reduction of the shot.
<b>colorSpace</b>	<a href="#">ColorSpaceType</a>	Read-write. The colorspace for the image.
<b>contrast</b>	number [-50..100]	Read-write. The contrast of the shot.
<b>exposure</b>	number [-4.0..4.0]	Read-write. The exposure of the shot.
<b>greenHue</b>	number [-100..100]	Read-write. The green hue of the shot.
<b>greenSaturation</b>	number [-100..100]	Read-write. The green saturation of the shot.
<b>luminanceSmoothing</b>	number [0..100]	Read-write. The luminance smoothing of the shot.
<b>redHue</b>	number [-100..100]	Read-write. The red hue of the shot.
<b>redSaturation</b>	number [-100..100]	Read-write. The red saturation of the shot.
<b>resolution</b>	number [1..999]	Read-write. The resolution of the document in pixels per inch.
<b>saturation</b>	number [-100..100]	Read-write. The saturation of the shot.
<b>settings</b>	<a href="#">CameraRAWSettingsType</a>	Read-write. The global settings for all Camera RAW options. Default: <code>CameraRAWSettingsType.CAMERA</code> .
<b>shadows</b>	number [0..100]	Read-write. The shadows of the shot.
<b>shadowTint</b>	number [-100..100]	Read-write. The shadow tint of the shot.
<b>sharpness</b>	number [0..100]	Read-write. The sharpness of the shot.
<b>size</b>	<a href="#">CameraRAWSize</a>	Read-write. The size of the new document.
<b>temperature</b>	number [2000..50000]	Read-write. The temperature of the shot.

Property	Value type	What it is (Continued)
<b>tint</b>	number [-150..150]	Read-write. The tint of the shot.
<b>typename</b>	string	Read-only. The class name of the referenced <code>cameraRAWOpenOptions</code> object.
<b>vignettingAmount</b>	number [-100..100]	Read-write. The vignetting amount of the shot.
<b>vignettingMidpoint</b>	number [-100..100]	Read-write. The vignetting mid point of the shot.
<b>whiteBalance</b>	<a href="#">WhiteBalanceType</a>	Read-write. The white balance options for the image. These are lighting conditions that affect color balance.



# Channel

Information about a color element in the image.

Access through the [Document.channels](#) collection. You can access an individual channel object in this list by index or by name. For example, this accesses a channel object in the active document by name and assigns an `opacity` value:

```
var channelRef = app.activeDocument.channels.getByNames("my channel");
channelRef.opacity = 22;
```

A channel is analogous to a plate in the printing process that applies a single color. The document's color mode determines the number of default channels; for example, an RGB document has three channels, red, green, and blue. A color can also have an alpha channel, which stores selections as masks, or a spot channel, which stores spot colors.

## Properties

Property	Value type	What it is
<b>color</b>	<a href="#">SolidColor</a>	Read-write. The color of the channel. Not valid when <code>kind</code> = <code>ChannelType.COMPONENT</code> .
<b>histogram</b>	array of number	Read-only. A histogram of the color of the channel. The array contains 256 members. Not valid when <code>kind</code> = <code>ChannelType.COMPONENT</code> . For component channel histogram values, use the <a href="#">histogram</a> property of the <a href="#">Document</a> object instead.
<b>kind</b>	<a href="#">ChannelType</a>	Read-write. The type of the channel.
<b>name</b>	string	Read-write. The name of the channel.
<b>opacity</b>	number [0..100]	Read-write. The opacity to use for alpha channels or the solidity to use for spot channels. Valid only when <code>kind</code> = <code>ChannelType.MASKEDAREA</code> or <code>SELECTEDAREA</code> .
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>typename</b>	string	Read-only. The class name of the referenced <code>channel</code> object.
<b>visible</b>	boolean	Read-write. True if the channel is visible.

## Methods

Method	Parameter type	Returns	What it does
<b>duplicate</b> ( [targetDocument] )	<a href="#">Document</a>	<a href="#">Channel</a>	Duplicates the channel.

Method	Parameter type	Returns	What it does
<b>merge</b> ( )			Merges a spot channel into the component channels.
<b>remove</b> ( )			Deletes the channel.

# Channels

The collection of [Channel](#) objects in a document.

Access through the [Document.channels](#) collection property. For example:

```
var channelRef = app.activeDocument.channels.add()
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>channels</code> collection.
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>typename</b>	string	Read-only. The class name of the referenced <code>channels</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>add</b> ( )		<a href="#">Channel</a>	Creates a new channel object and adds it to this collection.
<b>getByName</b> (name)	string	<a href="#">Channel</a>	Get the first element in the <code>channels</code> collection with the provided name.
<b>removeAll</b> ( )			Removes all alpha channel objects from the <code>channels</code> collection.

## Channels sample script

The following script opens a file if one is not already open, and then writes a histogram report (`histogram.log`) for the channels in the active document.

**Note:** This script contains a switch construction that uses a `break` statement. The `break` statement requires an ending semicolon (;), as in the following sample:

```
break;
```

### Histogram.jsx

```
// Function to activate all the channels according to the documents mode
// Takes a document reference for input
function TurnOnDocumentHistogramChannels(inDocument) {

    // see how many channels we need to activate
    var visibleChannelCount = 0

    // based on the mode of the document
    switch (inDocument.mode) {

        case DocumentMode.BITMAP:
```

```
        case DocumentMode.GRAYSCALE:
        case DocumentMode.INDEXEDCOLOR:
            visibleChannelCount = 1
            break;

        case DocumentMode.DUOTONE:
            visibleChannelCount = 2
            break;

        case DocumentMode.RGB:
        case DocumentMode.LAB:
            visibleChannelCount = 3
            break;

        case DocumentMode.CMYK:
            visibleChannelCount = 4
            break;

        case DocumentMode.MULTICHANNEL:
        default:
            visibleChannelCount = inDocument.channels.length + 1
            break;
    }

    // now get the channels to activate into a local array
    var aChannelArray = new Array()

    // index for the active channels array
    var aChannelIndex = 0

    for(var channelIndex = 0; channelIndex < inDocument.channels.length;
        channelIndex++) {
        if (channelIndex < visibleChannelCount) {
            aChannelArray[aChannelIndex++] = inDocument.channels[channelIndex]
        }
    }

    // now activate them
    inDocument.activeChannels = aChannelArray
}

// Save the current preferences
var startRulerUnits = app.preferences.rulerUnits
var startTypeUnits = app.preferences.typeUnits
var startDisplayDialogs = app.displayDialogs

// Set Adobe Photoshop CC 2015 to use pixels and display no dialogs
app.preferences.rulerUnits = Units.PIXELS
app.preferences.typeUnits = TypeUnits.PIXELS
app.displayDialogs = DialogModes.NO

// if there are no documents open then try to open a sample file
if (app.documents.length == 0) {
    open(File(app.path + "/Samples/Fish.psd"))
}

// get a reference to the working document
var docRef = app.activeDocument
```

```
// create the output file
// first figure out which kind of line feeds we need
if ($.os.search(/windows/i) != -1) {
    fileLineFeed = "Windows"
} else {
    fileLineFeed = "Macintosh"
}

// create the output file accordingly
fileOut = new File("~/Desktop/Histogram.log")
fileOut.lineFeed = fileLineFeed
fileOut.open("w", "TEXT", "????")

// write out a header
fileOut.write("Histogram report for " + docRef.name)

// find out how many pixels I have
var totalCount = docRef.width.value * docRef.height.value

// more info to the out file
fileOut.write(" with a total pixel count of " + totalCount + "\n")

// channel indexer
var channelIndex = 0

// remember which channels are currently active
var myActiveChannels = app.activeDocument.activeChannels

// document histogram only works in these modes
if (docRef.mode == DocumentMode.RGB ||
    docRef.mode == DocumentMode.INDEXEDCOLOR ||
    docRef.mode == DocumentMode.CMYK) {

    // activate the main channels so we can get the documents histogram
    TurnOnDocumentHistogramChannels(docRef)

    // Output the documents histogram
    OutputHistogram(docRef.histogram, "Luminosity", fileOut)
}

// local reference to work from
var myChannels = docRef.channels

// loop through each channel and output the histogram
for (var channelIndex = 0; channelIndex < myChannels.length; channelIndex++) {

    // the channel has to be visible to get a histogram
    myChannels[channelIndex].visible = true

    // turn off all the other channels
    for (var secondaryIndex = 0; secondaryIndex < myChannels.length;
        secondaryIndex++) {
        if (channelIndex != secondaryIndex) {
            myChannels[secondaryIndex].visible = false
        }
    }

    // Use the function to dump the histogram
    OutputHistogram(myChannels[channelIndex].histogram,
        myChannels[channelIndex].name, fileOut)
```

```
}

// close down the output file
fileOut.close()
alert("Histogram file saved to: " + fileOut.fsName)

// reset the active channels
docRef.activeChannels = myActiveChannels

// Reset the application preferences
app.preferences.rulerUnits = startRulerUnits
app.preferences.typeUnits = startTypeUnits
app.displayDialogs = startDisplayDialogs

// Utility function that takes a histogram and name
// and dumps to the output file
function OutputHistogram(inHistogram, inHistogramName, inOutFile) {

    // find out which count has the largest number
    // I scale everything to this number for the output
    var largestCount = 0

    // a simple indexer I can reuse
    var histogramIndex = 0

    // see how many samples we have total
    var histogramCount = 0

    // search through all and find the largest single item
    for (histogramIndex = 0; histogramIndex < inHistogram.length;
        histogramIndex++) {
        histogramCount += inHistogram[histogramIndex]
        if (inHistogram[histogramIndex] > largestCount)
            largestCount = inHistogram[histogramIndex]
    }

    // These should match
    if (histogramCount != totalCount) {
        alert("Something bad is happening!")
    }

    // see how much each "X" is going to count as
    var pixelsPerX = largestCount / 100

    // output this data to the file
    inOutFile.write("One X = " + pixelsPerX + " pixels.\n")

    // output the name of this histogram
    inOutFile.write(inHistogramName + "\n")

    // loop through all the items and output in the following format
    // 001
    // 002
    for (histogramIndex = 0; histogramIndex < inHistogram.length;
        histogramIndex++) {

        // I need an extra "0" for this line item to keep everything in line
        if (histogramIndex < 10)
            inOutFile.write("0")
```

```
// I need an extra "0" for this line item to keep everything in line
if (histogramIndex < 100)
    inOutFile.write("0")

// output the index to file
inOutFile.write(histogramIndex)

// some spacing to make it look nice
inOutFile.write(" ")

// figure out how many X's I need
var outputX = inHistogram[histogramIndex] / largestCount * 100

// output the X's
for (var a = 0; a < outputX; a++)
    inOutFile.write("X")

    inOutFile.write("\n")
}

inOutFile.write("\n")
}
```

# CMYKColor

Defines a CMYK color, used in the [SolidColor](#) object.

See also [GrayColor](#), [HSBColor](#), [LabColor](#), [NoColor](#), [RGBColor](#)

## Properties

Property	Value type	What it is
<b>black</b>	number [0.0..100.00]	Read-write. The black color value (as percent).
<b>cyan</b>	number [0.0..100.00]	Read-write. The cyan color value (as percent).
<b>magenta</b>	number [0.0..100.00]	Read-write. The magenta color value (as percent).
<b>typename</b>	string	Read-only. The class name of the referenced <code>CMYKColor</code> object.
<b>yellow</b>	number [0.0..100.00]	Read-write. The yellow color value (as percent).



# ColorSampler

A color sampler for a document. Access through the [Document.colorSamplers](#) collection. For example:

```
var colorSamplerRef = app.activeDocument.colorSamplers[0];  
var currentColor = colorSamplerRef.color;
```

**Note:** For additional information about color samplers, see Adobe Photoshop CC 2015 help on the Color Sampler Tool.

## Properties

Property	Value type	What it is
<b>color</b>	<a href="#">SolidColor</a>	Read-only. The color of the color sampler.
<b>position</b>	array of <a href="#">UnitValue</a>	Read-only. The position of the color sampler in the document. The array (x,y) represents the horizontal and vertical location of the count item.
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>typename</b>	string	Read-only. The class name of the referenced <code>ColorSampler</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>move</b> (position)	array of <a href="#">UnitValue</a>		Moves the color sampler to a new location in the document.  The <code>position</code> parameter (x,y) represents the new horizontal and vertical locations of the moved color sampler.
<b>remove</b> ( )			Deletes the <code>ColorSampler</code> object.

# ColorSamplers

The collection of [ColorSampler](#) objects in a document. Access through the [Document.colorSamplers](#) collection property. For example:

```
app.activeDocument.colorSamplers.removeAll()
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>ColorSamplers</code> collection.
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>typename</b>	string	Read-only. The class name of the referenced <code>ColorSamplers</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>add</b> (position)	array of <a href="#">UnitValue</a>	<a href="#">ColorSampler</a>	Creates a new color sampler object and adds it to this collection.  The <code>position</code> parameter (x,y) represents the new horizontal and vertical locations of the moved color sampler.
<b>removeAll</b> ( )			Removes all <code>ColorSampler</code> objects from the <code>ColorSamplers</code> collection.

# ContactSheetOptions

Options for creating a contact sheet with the [Application.makeContactSheet\(\)](#) method.

## Properties

Property	Value type	What it is
<b>acrossFirst</b>	boolean	Read-write. True to place the images horizontally (left to right, then top to bottom) first (default: <code>true</code> ).
<b>bestFit</b>	boolean	Read-write. True to rotate images for the best fit (default: <code>false</code> ).
<b>caption</b>	boolean	Read-write. True to use the filename as a caption for the image (default: <code>true</code> ).
<b>columnCount</b>	number [1..100]	Read-write. The number of columns to include (default: 5).
<b>flatten</b>	boolean	Read-write. True to flatten all layers in the final document (default: <code>true</code> ).
<b>font</b>	<a href="#">GalleryFontType</a>	Read-write. The font used for the caption (default: <code>GalleryFontType.ARIAL</code> ).
<b>fontSize</b>	number	Read-write. The font size to use for the caption (default: 12).
<b>height</b>	number [0..29000]	Read-write. The height (in pixels) of the resulting document (default: 720).
<b>horizontal</b>	number	Read-write. The horizontal spacing (in pixels) between images (default: 1).
<b>mode</b>	<a href="#">NewDocumentMode</a>	Read-write. The document color mode (default: <code>NewDocumentMode.RGB</code> ).
<b>resolution</b>	number [35..1200]	Read-write. The resolution of the document in pixels per inch (default: 72.0).
<b>rowCount</b>	number [1..100]	Read-write. The number of rows to use (default: 6).
<b>typename</b>	string	Read-only. The class name of the referenced <code>contactSheetOptions</code> object.
<b>useAutoSpacing</b>	boolean	Read-write. True to auto space the images (default: <code>true</code> ).
<b>vertical</b>	number [0..29000]	Read-write. The vertical spacing (in pixels) between images (default: 1). Valid only when <a href="#">useAutoSpacing</a> = <code>false</code> .
<b>width</b>	number [100..29000]	Read-write. The width (in pixels) of the resulting document (default: 576).

## CountItem

A counted item in a document. Access through the [Document.countItems](#) collection. See the [Document.autoCount\(\)](#) method.

**Note:** This feature is available in the Extended Version only.

For additional information about count items, see Adobe Photoshop CC 2015 help on the Count Tool.

### Properties

Property	Value type	What it is
<b>position</b>	array of <a href="#">UnitValue</a>	Read-only. The position of the count item in the document.
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>typename</b>	string	Read-only. The class name of the referenced <code>CountItem</code> object.

### Methods

Method	Parameter type	Returns	What it does
<b>remove</b> ( )			Deletes the <code>CountItem</code> object.

## CountItems

The collection of [CountItem](#) objects in the document.

Access through the [Document.countItems](#) collection property. For example:

```
app.activeDocument.countItems.removeAll()
```

**Note:** This feature is available in the Extended Version only.

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>CountItems</code> collection.
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>typename</b>	string	Read-only. The class name of the referenced <code>CountItems</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>add</b> (position)	array of <a href="#">UnitValue</a>	<a href="#">CountItem</a>	Creates a new count item object and adds it to this collection.  Parameter <code>position (x,y)</code> represents the horizontal and vertical positions, respectively, of the <code>CountItem</code> object.
<b>getByName</b> (name)	string	<a href="#">CountItem</a>	Get the first element in the <code>CountItems</code> collection with the provided name.
<b>removeAll</b> ( )			Removes all <code>CountItem</code> objects from the <code>CountItems</code> collection.

## DCS1\_SaveOptions

Options for saving a CMYK document in DCS1 format using the [Document.saveAs\(\)](#) method.

### Properties

Property	Value type	What it is
<b>dcs</b>	<a href="#">DCSType</a>	Read-write. (default: <code>DCSType.COLORCOMPOSITE</code> ).
<b>embedColorProfile</b>	boolean	Read-write. True to embed the color profile in the document
<b>encoding</b>	<a href="#">SaveEncoding</a>	Read-write. The type of encoding to use for document (default: <code>SaveEncoding.BINARY</code> ).
<b>halftoneScreen</b>	boolean	Read-write. True to include halftone screen (default: <code>false</code> ).
<b>interpolation</b>	boolean	Read-write. True to use image interpolation (default: <code>false</code> )
<b>preview</b>	<a href="#">Preview</a>	Read-write. The type of preview (default: <code>Preview.MACOSEIGHTBIT</code> ).
<b>transferFunction</b>	boolean	Read-write. True to include the Transfer functions to compensate for dot gain between the image and film (default: <code>false</code> ).
<b>typename</b>	string	Read-only. The class name of the referenced <code>DCS1_SaveOptions</code> object.
<b>vectorData</b>	boolean	Read-write. True to include vector data.  Valid only if the document includes vector data (unrasterized text).

## DCS2\_SaveOptions

Options for saving a CMYK document in DCS2 format using the [Document.saveAs\(\)](#) method.

### Properties

Property	Value type	What it is
<b>dcs</b>	<a href="#">DCSType</a>	Read-write. The type of composite file to create (default: <code>DCSType.NOCOMPOSITE</code> ).
<b>embedColorProfile</b>	boolean	Read-write. True to embed the color profile in the document.
<b>encoding</b>	<a href="#">SaveEncoding</a>	Read-write. The type of encoding to use (default: <code>SaveEncoding.BINARY</code> ).
<b>halftoneScreen</b>	boolean	Read-write. True to include the halftone screen (default: <code>false</code> ).
<b>interpolation</b>	boolean	Read-write. True to use image interpolation (default: <code>false</code> ).
<b>multiFileDCS</b>	boolean	Read-write. True to save color channels as multiple files or a single file (default: <code>false</code> ).
<b>preview</b>	<a href="#">Preview</a>	Read-write. The preview type (default: <code>Preview.MACOSEIGHTBIT</code> ).
<b>spotColors</b>	boolean	Read-write. True to save spot colors.
<b>transferFunction</b>	boolean	Read-write. True to include the Transfer functions to compensate for dot gain between the image and film (default: <code>false</code> ).
<b>typename</b>	string	Read-only. The class name of the referenced <code>DCS2_SaveOptions</code> object.
<b>vectorData</b>	boolean	Read-write. True to include vector data.  Valid only if the document includes vector data (unrasterized text).

## DICOMOpenOptions

Options for opening a document in DICOM format using the [Application.open\(\)](#) method.

**Note:** This feature is available in the Extended Version only.

### Properties

Property	Value type	What it is
<b>anonymize</b>	boolean	Read-write. True to make the patient information anonymous.
<b>columns</b>	number	Read-write. Number of columns in n-up configuration.
<b>reverse</b>	boolean	Read-write. True to reverse (invert) the image.
<b>rows</b>	number	Read-write. The number of rows in n-up configuration.
<b>showOverlays</b>	boolean	Read-write. True to show overlays.
<b>typename</b>	string	Read-only. The class name of the referenced <code>DICOMOpenOptions</code> object.
<b>windowLevel</b>	number	Read-write. The contrast of the image in Hounsfield units.
<b>windowWidth</b>	number	Read-write. The brightness of the image in Hounsfield units.



# Document

The active containment object for layers and all other objects in the script; the basic canvas for the file.

- Access the object for the currently active document through [Application.activeDocument](#).
- You can access other documents, or iterate through all open documents using the list in the [Application.documents](#) collection. You can access individual documents in the list by index, or use [Documents.getByIndex\(\)](#) to retrieve them by name.
- Create documents programmatically using the [Documents.add\(\)](#) method.

See [Document sample script](#) and the [Documents](#) collection object for examples.

**Note:** In Adobe Photoshop CC 2015, a document can also be referred to as an image or a canvas.

- The term *image* refers to the entire document and its contents. You can trim or crop an image. You resize an image using the [resizeImage\(\)](#) method.
- The term *canvas* refers to the space in which the document sits on the screen. You can rotate or flip the canvas. You resize the canvas using the [resizeCanvas\(\)](#) method.

## Properties

Property	Value type	What it is
<b>activeChannels</b>	array of <a href="#">Channel</a>	Read-write. The selected channels.
<b>activeHistoryBrushSource</b>	<a href="#">Guide</a>	Read-write. The history state to use with the history brush.
<b>activeHistoryState</b>	<a href="#">Guide</a>	Read-write. The selected <code>HistoryState</code> object.
<b>activeLayer</b>	<a href="#">ArtLayer</a> or <a href="#">LayerSet</a>	Read-write. The selected layer.
<b>artLayers</b>	<a href="#">ArtLayers</a>	Read-only. The art layers collection.
<b>backgroundLayer</b>	<a href="#">ArtLayer</a>	Read-only. The background layer of the document.
<b>bitsPerChannel</b>	<a href="#">BitsPerChannelType</a>	Read-write. The number of bits per channel.
<b>channels</b>	<a href="#">Channels</a>	Read-only. The channels collection.
<b>colorProfileName</b>	string	Read-write. The name of the color profile.  Valid only when <a href="#">colorProfileType</a> = <code>ColorProfile.CUSTOM OR WORKING</code> .
<b>colorProfileType</b>	<a href="#">ColorProfileType</a>	Read-write. Whether the document uses the working color profile, a custom profile, or no profile.
<b>colorSamplers</b>	<a href="#">ColorSamplers</a>	Read-only. The current color samplers associated with this document.
<b>componentChannels</b>	array of <a href="#">Channel</a>	Read-only. The color channels that make up the document; for instance, the Red, Green, and Blue channels for an RGB document.

Property	Value type	What it is (Continued)
<code>countItems</code>	<a href="#">CountItems</a>	Read-only. The current count items.  <b>Note:</b> For additional information about count items, see Adobe Photoshop CC 2015 help on the Count Tool.
<code>fullName</code>	<a href="#">File</a>	Read-only. The full path name of the document.
<code>guides</code>	<a href="#">Guides</a>	Read-only. The guides collection.
<code>height</code>	<a href="#">UnitValue</a>	Read-only. The height of the document (unit value).
<code>histogram</code>	array of number	Read-only. A histogram showing the number of pixels at each color intensity level for the composite channel. The array contains 256 members.  Valid only when <code>mode</code> = <code>DocumentMode.RGB</code> , <code>CMYK</code> ; or <code>INDEXEDCOLOR</code> .
<code>historyStates</code>	<a href="#">HistoryStates</a>	Read-only. The history states collection.
<code>info</code>	<a href="#">DocumentInfo</a>	Read-only. Metadata about the document.
<code>layerComps</code>	<a href="#">LayerComps</a>	Read-only. The layer compositions collection.
<code>layers</code>	<a href="#">Layers</a>	Read-only. The layers collection.
<code>layerSets</code>	<a href="#">LayerSets</a>	Read-only. The layer set collection.
<code>managed</code>	boolean	Read-only. True if the document is a workgroup document.
<code>measurementScale</code>	<a href="#">MeasurementScale</a>	Read-only. The measurement scale for the document.  <b>Note:</b> The measurement scale feature is available in the Extended version only.
<code>mode</code>	<a href="#">DocumentMode</a>	Read-only. The color profile.
<code>name</code>	string	Read-only. The document's name.
<code>parent</code>	<a href="#">Application</a>	Read-only. The application object that contains this document.
<code>path</code>	<a href="#">File</a>	Read-only. The path to the document.
<code>pathItems</code>	<a href="#">PathItems</a>	Read-only. The path items collection.
<code>pixelAspectRatio</code>	number [0.100..10.000]	Read-write. The (custom) pixel aspect ratio to use.
<code>printSettings</code>	<a href="#">DocumentPrintSettings</a>	Read-only. The print settings for the document.

Property	Value type	What it is (Continued)
<b>quickMaskMode</b>	boolean	Read-write. True if the document is in Quick Mask mode.
<b>resolution</b>	number	Read-only. The document's resolution (in pixels per inch).
<b>saved</b>	boolean	Read-only. True if the document has been saved since the last change.
<b>selection</b>	<a href="#">Selection</a>	Read-only. The selected area of the document.
<b>typename</b>	string	Read-only. The class name of the Document object.
<b>width</b>	<a href="#">UnitValue</a>	Read-only. The width of the document (unit value).
<b>xmpMetadata</b>	<a href="#">xmpMetadata</a>	Read-only. XMP metadata for the document. Camera RAW settings for the image are stored here for example.

## Methods

Method	Parameter type	Returns	What it does
<b>autoCount</b> (channel, threshold)	<a href="#">Channel</a> number		<p>Counts the number of objects in a document. Available in the Extended Version only.</p> <p>Creates a <a href="#">CountItem</a> object for each object counted.</p> <p>For additional information about how to set up objects to count, see the Count Tool in the Adobe Photoshop CC 2015 Help</p>
<b>changeMode</b> (destinationMode [, options])	<a href="#">ChangeMode</a> <a href="#">BitmapConversionOptions</a> or <a href="#">IndexedConversionOptions</a>		<p>Changes the color profile of the document.</p>
<b>close</b> ([saving])	<a href="#">SaveOptions</a>		<p>Closes the document. If any changes have been made, the script presents an alert with three options: save, do not save, prompt to save. The optional parameter specifies a selection in the alert box (default: <code>SaveOptionsType.PROMPTTOSAVECHANGES</code>).</p>
<b>convertProfile</b> (destinationProfile, intent [, blackPointCompensation] [, dither])	string <a href="#">Intent</a> boolean boolean		<p>Changes the color profile.</p> <p>The <code>destinationProfile</code> parameter must be either a string that names the color mode or <code>Working RGB</code>, <code>Working CMYK</code>, <code>Working Gray</code>, <code>Lab Color</code> (meaning one of the working color spaces or Lab color).</p>
<b>crop</b> (bounds [, angle] [, width] [, height])	array of 4 <a href="#">UnitValue</a> number <a href="#">UnitValue</a> <a href="#">UnitValue</a>		<p>Crops the document. The <code>bounds</code> parameter is an array of four coordinates for the region remaining after cropping, [left, top, right, bottom].</p>
<b>duplicate</b> ([name ] [, mergeLayersOnly] ) )	string boolean	<a href="#">Document</a>	<p>Creates a duplicate of the document object.</p> <p>The optional parameter <code>name</code> provides the name for the duplicated document.</p> <p>The optional parameter <code>mergeLayersOnly</code> indicates whether to only duplicate merged layers.</p>

Method	Parameter type	Returns	What it does (Continued)
<b>exportDocument</b> (exportIn [, exportAs] [, options])	<a href="#">File</a> <a href="#">ExportType</a> <a href="#">ExportOptionsIllustrator</a> —or— <a href="#">ExportOptionsSaveForWeb</a>		Exports the paths in the document to an Illustrator file, or exports the document to a file with Web or device viewing optimizations.  This is equivalent to choosing <b>File &gt; Export &gt; Paths To Illustrator</b> , or <b>File &gt; Save For Web and Devices</b> .
<b>flatten</b> ()			Flattens all layers in the document.
<b>flipCanvas</b> (direction)	<a href="#">Direction</a>		Flips the image within the canvas in the specified direction.
<b>importAnnotations</b> (file)	<a href="#">File</a>		Imports annotations into the document.
<b>mergeVisibleLayers</b> ()			Flattens all visible layers in the document.
<b>paste</b> ([intoSelection])	boolean	<a href="#">ArtLayer</a>	Pastes the contents of the clipboard into the document. If the optional argument is set to <code>true</code> and a selection is active, the contents are pasted into the selection.
<b>print</b> ([sourceSpace] [, printSpace] [, intent] [blackPointCompensation])	<a href="#">SourceSpaceType</a> string <a href="#">Intent</a> boolean		Prints the document.  <code>printSpace</code> specifies the color space for the printer. Valid values are <code>nothing</code> (that is, the same as the source); or <code>Working RGB</code> , <code>Working CMYK</code> , <code>Working Gray</code> , <code>Lab Color</code> (meaning one of the working color spaces or Lab color); or a string specifying a specific colorspace (default is same as source).
<b>printOneCopy</b> ()			Print one copy of the document.
<b>rasterizeAllLayers</b> ()			Rasterizes all layers.
<b>recordMeasurements</b> ([source] [, dataPoints])	<a href="#">MeasurementSource</a> array of string		Record measurements of document.
<b>resizeCanvas</b> ([width] [, height] [, anchor])	<a href="#">UnitValue</a> <a href="#">UnitValue</a> <a href="#">AnchorPosition</a>		Changes the size of the canvas to display more or less of the image but does not change the image size. See <a href="#">resizeImage</a> .

Method	Parameter type	Returns	What it does (Continued)
<b>resizeImage</b> ([width] [, height] [, resolution] [, resampleMethod] [, amount])	<a href="#">UnitValue</a> <a href="#">UnitValue</a> number <a href="#">ResampleMethod</a> number		Changes the size of the image.  The <code>amount</code> parameter controls the amount of noise value when using preserve details (Range: 0 - 100).
<b>revealAll</b> ()			Expands the document to show clipped sections.
<b>rotateCanvas</b> (angle)	number		Rotates the canvas (including the image) in clockwise direction.
<b>save</b> ()			Saves the document.
<b>saveAs</b> (saveIn [, options] [, asCopy] [, extensionType])	<a href="#">File</a> object (see description) boolean <a href="#">Extension</a>		Saves the document in a specific format.  Specify the save options appropriate to the format by passing one of these objects:  <a href="#">BMPSaveOptions</a> <a href="#">DCS1_SaveOptions</a> <a href="#">DCS2_SaveOptions</a> <a href="#">EPSSaveOptions</a> <a href="#">GIFSaveOptions</a> <a href="#">JPEGSaveOptions</a> <a href="#">PDFSaveOptions</a> <a href="#">PhotoshopSaveOptions</a> <a href="#">PCTFileSaveOptions</a> <a href="#">PCTResourceSaveOptions</a> <a href="#">PixarSaveOptions</a> <a href="#">PNGSaveOptions</a> <a href="#">RawSaveOptions</a> <a href="#">SGIRGBSaveOptions</a> <a href="#">TargaSaveOptions</a> <a href="#">TiffSaveOptions</a>
<b>splitChannels</b> ()		array of <a href="#">Document</a>	Splits the document channels into separate images.
<b>suspendHistory</b> (historyString javascriptString)	string string		Provides a single entry in history states for the entire script provided by <code>javascriptString</code> . Allows a single undo for all actions taken in the script.  The <code>historyString</code> parameter provides the string to use for the history state.  The <code>javascriptString</code> parameter provides a string of JavaScript code to execute while history is suspended.
<b>trap</b> (width)	number		Applies trapping to a CMYK document.  Valid only when <code>docRef.<a href="#">mode</a> = DocumentMode.CMYK</code> .

Method	Parameter type	Returns	What it does (Continued)
<b>trim</b> ([type] [, top] [, left] [, bottom] [, right])	<a href="#">TrimType</a> boolean boolean boolean boolean		Trims the transparent area around the image on the specified sides of the canvas.  Default is <code>true</code> for all Boolean parameters.

## Document sample script

The following script creates a document that contains two images (a sunflower and a duck) obtained from the Adobe Photoshop CC 2015 Samples folder and employs the following steps:

- Determines which image is larger.
- Resizes the smaller image to match the larger image.
- Creates a merged document twice as high as either image in order to hold both images.
- Selects part of the document and pastes the sunflower into the selection.
- Inverts the selection and pastes the duck into the lower part of the document.
- Positions the sunflower over the duck.

### Document.jsx

```
// Save the current preferences
var startRulerUnits = app.preferences.rulerUnits
var startTypeUnits = app.preferences.typeUnits
var startDisplayDialogs = app.displayDialogs

// Set Adobe Photoshop CC 2015 to use pixels and display no dialogs
app.preferences.rulerUnits = Units.PIXELS
app.preferences.typeUnits = TypeUnits.PIXELS
app.displayDialogs = DialogModes.NO

// first close all the open documents
while (app.documents.length) {
    app.activeDocument.close()
}

// Open the sunflower and duck files from the samples folder
var flowerDoc = open(File(app.path + "/Samples/Sunflower.psd"))
var duckDoc = open(File(app.path + "/Samples/Ducky.tif"))

// Find out which document is larger
// Resize the smaller document the to the larger document's size
// The resize requires the document be the active/front document
if ((flowerDoc.width.value * flowerDoc.height.value) >
    (duckDoc.width.value * duckDoc.height.value)) {
    app.activeDocument = duckDoc
    duckDoc.resize(flowerDoc.width, flowerDoc.height)
}
else {
    app.activeDocument = flowerDoc
    flowerDoc.resizeImage(duckDoc.width, duckDoc.height)
}

// Create a new document twice as high as two files
```

```
var mergedDoc = app.documents.add(duckDoc.width, duckDoc.height * 2,
duckDoc.resolution, "FlowerOverDuck")

// Copy the flower to the top; make it the active document so we can manipulate it
app.activeDocument = flowerDoc
flowerDoc.activeLayer.copy()

//Paste the flower to the merged document, making the merged document active
app.activeDocument = mergedDoc

// Select a square area at the top of the new document
var selRegion = Array(Array(0, 0),
    Array(mergedDoc.width.value, 0),
    Array(mergedDoc.width.value, mergedDoc.height.value / 2),
    Array(0, mergedDoc.height.value / 2),
    Array(0, 0))
// Create the selection
mergedDoc.selection.select(selRegion)

//Paste in the flower
mergedDoc.paste(TRUE)

// do the same thing for the duck
app.activeDocument = duckDoc
duckDoc.activeLayer.copy()

app.activeDocument = mergedDoc
mergedDoc.selection.select(selRegion)

// Inverting the selection so the bottom of the document is now selected
mergedDoc.selection.invert()

// Paste the duck
mergedDoc.paste(TRUE)

// get rid of our originals without modifying them
duckDoc.close(SaveOptions.DONOTSAVECHANGES)
flowerDoc.close(SaveOptions.DONOTSAVECHANGES)

// Reset the application preferences
app.preferences.rulerUnits = startRulerUnits
app.preferences.typeUnits = startTypeUnits
app.displayDialogs = startDisplayDialogs
```



# DocumentPrintSettings

The print settings for a document.

## Properties

Property	Value type	What it is
<b>backgroundColor</b>	<a href="#">SolidColor</a>	Read-write. Background color of page.
<b>bleedWidth</b>	<a href="#">UnitValue</a>	Read-write. Bleed width
<b>caption</b>	boolean	Read-write. Print the caption found in FileInfo.
<b>centerCropMarks</b>	boolean	Read-write. Print center crop marks.
<b>colorBars</b>	boolean	Read-write. Print color calibration bars.
<b>copies</b>	number	Read-write. Number of copies to print.
<b>cornerCropMarks</b>	boolean	Read-write. Print corner crop marks.
<b>colorHandling</b>	<a href="#">PrintColorHandling</a>	Read-only. Color handling.
<b>activePrinter</b>	string	Read-write. The currently active printer.
<b>flip</b>	boolean	Read-write. Flip the image horizontally.
<b>hardProof</b>	boolean	Read-write. Print a hard proof.
<b>interpolate</b>	boolean	Read-write.
<b>labels</b>	boolean	Read-write. Prints the document title.
<b>mapBlack</b>	boolean	Read-write. Map blacks.
<b>negative</b>	boolean	Read-write. Invert the image colors.
<b>renderIntent</b>	Intent	Read-write. Color conversion intent when print space is different from the source space.
<b>posX</b>	<a href="#">UnitValue</a>	Read-only. The x position of the image on page.
<b>posY</b>	<a href="#">UnitValue</a>	Read-only. The y position of the image on page.
<b>printBorder</b>	<a href="#">UnitValue</a>	Read-write. The width of the print border.
<b>printerName</b>	string	Read-write. Name of the printer.
<b>printSpace</b>	string	Read-write. color space for printer. Can be nothing (meaning same as source); 'Working RGB', 'Working CMYK', 'Working Gray', 'Lab Color' (meaning one of the working spaces or Lab color); or a string specifying a specific colorspace (default is same as source)
<b>registrationMarks</b>	boolean	Read-write. Print registration marks.

Property	Value type	What it is (Continued)
<b>scale</b>	number	Read-only. Scale of image on page.
<b>vectorData</b>	boolean	Read-write. Include vector data.

## Methods

Method	Parameter type	Returns	What it does
<b>setPagePosition</b> (docPosition, posX, posY, scale)	<a href="#">DocPositionStyle</a> <a href="#">UnitValue</a> <a href="#">UnitValue</a> number		Set the position of the image on the page.

## DocumentInfo

Metadata about a document object.

Access through the [Document.info](#) property. For example, the following sets the author, caption, and copyrighted properties:

```
var docRef = open(fileList[i])
// set the file info
docRef.info.author = "Mr. Adobe programmer"
docRef.info.caption = "Adobe Photo shoot"
docRef.info.copyrighted = CopyrightedType.COPYRIGHTEDWORK
```

These values can be set interactively by choosing **File > File Info**.

## Properties

Property	Value type	What it is
<b>author</b>	string	Read-write.
<b>authorPosition</b>	string	Read-write.
<b>caption</b>	string	Read-write.
<b>captionWriter</b>	string	Read-write.
<b>category</b>	string	Read-write.
<b>city</b>	string	Read-write.
<b>copyrighted</b>	<a href="#">CopyrightedType</a>	Read-write. The copyrighted status.
<b>copyrightNotice</b>	string	Read-write.
<b>country</b>	string	Read-write.
<b>creationDate</b>	string	Read-write.
<b>credit</b>	string	Read-write.
<b>exif</b>	array of array [ <i>tag data</i> ]	Read-only. Camera data that includes camera settings used when the image was taken.  Each array member is a tag pair, an array of [ <i>tag</i> , <i>tag_data</i> ]; for example, [ "camera" "Cannon"].
<b>headline</b>	string	Read-write.
<b>instructions</b>	string	Read-write.
<b>jobName</b>	string	Read-write.
<b>keywords</b>	array of string	Read-write. A list of keywords that can identify the document or its contents.
<b>ownerUrl</b>	string	Read-write.

Property	Value type	What it is (Continued)
<b>parent</b>	<a href="#">Document</a>	Read-only. The <code>info</code> object's container.
<b>provinceState</b>	string	Read-write.
<b>source</b>	string	Read-write.
<b>supplementalCategories</b>	array of string	Read-write.
<b>title</b>	string	Read-write.
<b>transmissionReference</b>	string	Read-write.
<b>typename</b>	string	Read-only. The class name of the referenced <code>info</code> object.
<b>urgency</b>	<a href="#">Urgency</a>	Read-write.

## DocumentInfo sample Script

The following script sets document info (metadata) for all of the files in a specified folder and then saves the modified files as low-quality JPEG images in a new folder without changing the originals.

- Ask the user to specify the folder that contains the original files and the output folder for the JPEG images, and then check that the folders exist.
- Open each file and use the `documentInfo` object properties to tag it with the following metadata:
  - author: Adobe programmer
  - caption: Adobe Photo shoot
  - captionWriter: Adobe programmer
  - city: San Jose
  - copyrightNotice: Copyright (c) Adobe programmer Photography
  - copyrighted status: Copyrighted Work
  - country: USA
  - state: CA
- Save the new documents in JPEG format with a low quality setting.

### DocumentInfo.jsx

```
// Save the current preferences
var startDisplayDialogs = app.displayDialogs

// Set Adobe Photoshop CC 2015 to use pixels and display no dialogs
app.displayDialogs = DialogModes.NO

// ask the user for the input and output folders
var inputFolder = Folder.selectDialog("Select a folder to tag")
var outputFolder = Folder.selectDialog("Select a folder for the output files")

// see if we got something interesting from the dialog
if (inputFolder != null && outputFolder != null) {
    // get all the files found in this folder
    var fileList = inputFolder.GetFiles()
```

```
// save the outputs in JPEG
var jpegOptions = new JPEGSaveOptions()
// set the jpeg quality really low so the files are small
jpegOptions.quality = 1
// open each one in turn
for (var i = 0; i < fileList.length; i++) {
    // The fileList includes both folders and files so open only files
    if (fileList[i] instanceof File && fileList[i].hidden == false) {
        // get a reference to the new document
        var docRef = open(fileList[i])

        // tag all of the documents with photo shoot information
        docRef.info.author = "Adobe programmer"
        docRef.info.caption = "Adobe Photo shoot"
        docRef.info.captionWriter = "Adobe programmer"
        docRef.info.city = "San Jose"
        docRef.info.copyrightNotice = "Copyright (c) Adobe programmer
            Photography"
        docRef.info.copyrighted = CopyrightedType.COPYRIGHTEDWORK
        docRef.info.country = "USA"
        docRef.info.provinceState = "CA"

        // change the date to a Adobe Photoshop CC 2015 date format
        // "YYYYMMDD"
        var theDate = new Date()
        // the year is from 1900 ???
        var theYear = (theDate.getYear() + 1900).toString()
        // convert the month from 0..12 to 00..12
        var theMonth = theDate.getMonth().toString()
        if (theDate.getMonth() < 10) {
            theMonth = "0" + theMonth
        }
        // convert the day from 0..31 to 00..31
        var theDay = theDate.getDate().toString()
        if (theDate.getDate() < 10) {
            theDay = "0" + theDay
        }
        // stick them all together
        docRef.info.creationDate = theYear + theMonth + theDay

        // flatten because we are saving to JPEG
        docRef.flatten()
        // go to 8 bit because we are saving to JPEG
        docRef.bitsPerChannel = BitsPerChannelType.EIGHT

        // save and close
        docRef.saveAs(new File(outputFolder + "/Output" + i + ".jpg"), jpegOptions)

        // don't modify the original
        docRef.close(SaveOptions.DONOTSAVECHANGES)
    }
}

// Reset the application preferences
app.displayDialogs = startDisplayDialogs
```

## Documents

The collection of open [Document](#) objects.

Access this list through the [Application.documents](#) collection property, which is available through the `app` global variable, or directly at the top level. For example, the following adds a new document to the collection:

```
app.documents.add(800, 500, 72, "myDocument", NewDocumentMode.RGB)
```

—or—

```
documents.add(800, 500, 72, "myDocument", NewDocumentMode.RGB)
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>documents</code> collection.
<b>parent</b>	<a href="#">Application</a>	Read-only. The containing application.
<b>typename</b>	string	Read-only. The class name of the referenced <code>documents</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>add</b> ( [width] [, height] [, resolution] [, name] [, mode] [, initialFill] [, pixelAspectRatio] [, bitsPerChannel]  [, colorProfileName] )	<a href="#">UnitValue</a> <a href="#">UnitValue</a> number string <a href="#">NewDocumentMode</a> <a href="#">DocumentFill</a> number [0.1..10.00] <a href="#">BitsPerChannelType</a> string	<a href="#">Document</a>	Creates a new document object and adds it to this collection.  <code>pixelAspectRatio</code> : Default is 1.0, a square aspect ratio.  <code>bitsPerChannelType</code> : Default is <code>BitsPerChannelType.EIGHT</code> .
<b>getByName</b> (name)	string	<a href="#">Document</a>	Gets the first element in the <code>documents</code> collection with the provided name

## EPSOpenOptions

Options for opening a document in EPS format using the [Application.open\(\)](#) method.

### Properties

Property	Value type	What it is
<b>antiAlias</b>	boolean	Read-write. True to use antialias.
<b>constrainProportions</b>	boolean	Read-write. True to constrain the proportions of the image.
<b>height</b>	<a href="#">UnitValue</a>	Read-write. The height of the image (unit value).
<b>mode</b>	<a href="#">OpenDocumentMode</a>	Read-write. The color profile to use as the document mode.
<b>resolution</b>	number	Read-write. The resolution of the document in pixels per inch.
<b>typename</b>	string	Read-only. The class name of the referenced <code>EPSOpenOptions</code> object.
<b>width</b>	<a href="#">UnitValue</a>	Read-write. The width of the image (unit value).



## EPSSaveOptions

Options for saving a document in EPS format using the [Document.saveAs\(\)](#) method.

### Properties

Property	Value type	What it is
<b>embedColorProfile</b>	boolean	Read-write. True to embed the color profile in this document.
<b>encoding</b>	<a href="#">SaveEncoding</a>	Read-write. The type of encoding to use (default: <code>SaveEncoding.BINARY</code> ).
<b>halftoneScreen</b>	boolean	Read-write. True to include the halftone screen (default: <code>false</code> ).
<b>interpolation</b>	boolean	Read-write. True to use image interpolation (default: <code>false</code> ).
<b>preview</b>	<a href="#">Preview</a>	Read-write. The preview type.
<b>psColorManagement</b>	boolean	Read-write. True to use Postscript color management (default: <code>false</code> ).
<b>transferFunction</b>	boolean	Read-write. True to include the Transfer functions to compensate for dot gain between the image and film (default: <code>false</code> ).
<b>transparentWhites</b>	boolean	Read-write. True to display white areas as transparent.  Valid only when <code>document.mode</code> = <code>DocumentMode.BITMAP</code> . See also <a href="#">changeMode()</a> .
<b>typename</b>	string	Read-only. The class name of the referenced <code>EPSSaveOptions</code> object.
<b>vectorData</b>	boolean	Read-write. True to include vector data.  Valid only if the document includes vector data (text).

## ExportOptionsIllustrator

Options for exporting [PathItem](#) objects to an Adobe Illustrator® file using using the [Document.exportDocument\(\)](#) method. These options are the options that you can provide when you choose **File > Export > Paths To Illustrator**.

### Properties

Property	Value type	What it is
<b>path</b>	<a href="#">IllustratorPathType</a>	Read-write. The type of path to export (default: <code>IllustratorPathType.DOCUMENTBOUNDS</code> ).
<b>pathName</b>	string	Read-write. The name of the path to export. Valid only when <a href="#">path</a> = <code>IllustratorPathType.NAMEDPATH</code> .
<b>typename</b>	string	Read-only. The class name of the referenced <code>exportOptionsIllustrator</code> object.

## ExportOptionsSaveForWeb

Options for optimizing a document for the web or devices using the [Document.exportDocument\(\)](#) method. These are the options that you can provide when you choose **File > Save For Web and Devices**.

### Properties

Property	Value type	What it is
<b>blur</b>	number	Read-write. Applies blur to the image to reduce artifacts (default: 0.0).
<b>colorReduction</b>	<a href="#">ColorReductionType</a>	Read-write. The color reduction algorithm (default: <code>ColorReductionType.SELECTIVE</code> ).
<b>colors</b>	number	Read-write. The number of colors in the palette (default: 256).
<b>dither</b>	<a href="#">Dither</a>	Read-write. The type of dither (default: <code>Dither.DIFFUSION</code> ).
<b>ditherAmount</b>	number	Read-write. The amount of dither (default: 100). Valid only when <a href="#">dither</a> = <code>Dither.DIFFUSION</code> .
<b>format</b>	<a href="#">SaveDocumentType</a>	Read-write. The file format to use (default: <code>SaveDocumentType.COMPUSEVEGIF</code> ).  <b>Note:</b> For this property, only <code>COMPUSEVEGIF</code> , <code>JPEG</code> , <code>PNG-8</code> , <code>PNG-24</code> , and <code>BMP</code> are supported.
<b>includeProfile</b>	boolean	Read-write. True to include the document's embedded color profile (default: <code>false</code> ).
<b>interlaced</b>	boolean	Read-write. True to download in multiple passes; progressive (default: <code>false</code> ).
<b>lossy</b>	number	Read-write. The amount of lossiness allowed (default: 0).
<b>matteColor</b>	<a href="#">RGBColor</a>	Read-write. The colors to blend transparent pixels against.
<b>optimized</b>	boolean	Read-write. True to create smaller but less compatible files (default: <code>true</code> ). Valid only when <a href="#">format</a> = <code>SaveDocumentType.JPEG</code> .
<b>PNG8</b>	boolean	Read-write. Indicates the number of bits; <code>true</code> = 8, <code>false</code> = 24 (default: <code>true</code> ). Valid only when <a href="#">format</a> = <code>SaveDocumentType.PNG</code> .

Property	Value type	What it is (Continued)
<b>quality</b>	number [0..100]	Read-write. The quality of the produced image as a percentage; default: 60.
<b>transparency</b>	boolean	Read-write. Indication of transparent areas of the image should be included in the saved image(default: true).
<b>transparencyAmount</b>	number	Read-write. The amount of transparency dither (default: 100). Valid only if <a href="#">transparency</a> = true.
<b>transparencyDither</b>	<a href="#">Dither</a>	Read-write. The transparency dither algorithm (default: transparencyDither = Dither.NONE).
<b>typename</b>	string	Read-only. The class name of the referenced ExportOptionsSaveForWeb object.
<b>webSnap</b>	number	Read-write. The tolerance amount within which to snap close colors to web palette colors (default: 0).

# File

## Folder

ExtendScript defines the JavaScript classes `File` and `Folder` to encapsulate file-system references in a platform-independent manner; see ['JavaScript support in Adobe Photoshop CC 2015' on page 32](#). For references details of these classes, see the *JavaScript Tools Guide*.

# GalleryBannerOptions

Options for the [bannerOptions](#) property of the [GalleryOptions](#) object.

**Tip:** You can preserve default values for many of these properties by setting the [GalleryOptions](#) property [preserveAllMetadata](#) to true; this is the same as choosing **File > Automate > Web Photo Gallery**, and then choosing **Preserve all metadata** in the Options area of the Web Photo Gallery dialog.

## Properties

Property	Value type	What it is
<code>contactInfo</code>	string	Read-write. The web photo gallery contact info.
<code>date</code>	string	Read-write. The web photo gallery date (default: current date).
<code>font</code>	<a href="#">GalleryFontType</a>	Read-write. The font setting for the banner text (default: <code>GalleryFontType.ARIAL</code> ).
<code>fontSize</code>	number [1..7]	Read-write. The font size for the banner text (default: 3).
<code>photographer</code>	string	Read-write. The web photo gallery photographer.
<code>siteName</code>	string	Read-write. The web photo gallery site name (default: <code>Adobe Web Photo Gallery</code> ).
<code>typename</code>	string	Read-only. The class name of the referenced <code>galleryBannerOptions</code> object.

## GalleryCustomColorOptions

Options for the [customColorOptions](#) property of the [GalleryOptions](#) object.

**Tip:** You can preserve default values for many of these properties by setting the [GalleryOptions](#) property [preserveAllMetadata](#) to true; this is the same as choosing **File > Automate > Web Photo Gallery**, and then choosing **Preserve all metadata** in the Options area of the Web Photo Gallery dialog.

### Properties

Property	Value type	What it is
<code>activeLinkColor</code>	<a href="#">RGBColor</a>	Read-write. The color to use to indicate an active link.
<code>backgroundColor</code>	<a href="#">RGBColor</a>	Read-write. The background color.
<code>bannerColor</code>	<a href="#">RGBColor</a>	Read-write. The banner color.
<code>linkColor</code>	<a href="#">RGBColor</a>	Read-write. The color to use to indicate a link.
<code>textColor</code>	<a href="#">RGBColor</a>	Read-write. The text color.
<code>typename</code>	string	Read-only. The class name of the referenced <code>galleryCustomColorOptions</code> object.
<code>visitedLinkColor</code>	<a href="#">RGBColor</a>	Read-write. The color to use to indicate a visited link.

# GalleryImagesOptions

Options for the [imagesOptions](#) property of the [GalleryOptions](#) object.

**Tip:** You can preserve default values for many of these properties by setting the [GalleryOptions](#) property [preserveAllMetadata](#) to true; this is the same as choosing **File > Automate > Web Photo Gallery**, and then choosing **Preserve all metadata** in the Options area of the Web Photo Gallery dialog.

## Properties

Property	Value type	What it is
<b>border</b>	number [0..99]	Read-write. The size (in pixels) of the border that separates images (default: 0).
<b>caption</b>	boolean	Read-write. True to generate image captions (default: false).
<b>dimension</b>	number	Read-write. The resized image dimensions in pixels (default: 350). Valid only when <a href="#">resizeImages</a> = true.
<b>font</b>	<a href="#">GalleryFontType</a>	Read-write. The font to use for image captions (default: <code>GalleryFontType.ARIAL</code> ).
<b>fontSize</b>	number [1..7]	Read-write. The font size for image captions (default: 3). Valid only when <a href="#">caption</a> = true.
<b>imageQuality</b>	number [0..12]	Read-write. The quality setting for a JPEG image (default: 5).
<b>includeCopyright</b>	boolean	Read-write. True to include copyright information in captions (default: false). Valid only when <a href="#">caption</a> = true.
<b>includeCredits</b>	boolean	Read-write. True to include the credits in image captions (default: false). Valid only when <a href="#">caption</a> = true.
<b>includeFilename</b>	boolean	Read-write. True to include the file name in image captions (default: true). Valid only when <a href="#">caption</a> = true.
<b>includeTitle</b>	boolean	Read-write. True to include the title in image captions (default: false). Valid only when <a href="#">caption</a> = true.
<b>numericLinks</b>	boolean	Read-write. True to add numeric links (default: true).



Property	Value type	What it is (Continued)
<b>resizeConstraint</b>	<a href="#">GalleryConstrainType</a>	Read-write. The image dimensions to constrain in the gallery image (default: <code>GalleryConstrainType.CONSTRAINBOTH</code> ). Valid only when <a href="#">resizeImages</a> = <code>true</code> .
<b>resizeImages</b>	boolean	Read-write. True to automatically resize images for placement on the gallery pages (default: <code>true</code> ).
<b>typename</b>	string	Read-only. The class name of the referenced <code>galleryImagesOptions</code> object.

# GalleryOptions

Options for a Web photo gallery, created with [Application.makePhotoGallery\(\)](#).

**Tip:** You can preserve default values for many of these properties by choosing **File > Automate > Web Photo Gallery**, and then choosing **Preserve all metadata** in the Options area of the Web Photo Gallery dialog.

## Properties

Property	Value type	What it is
<code>addSizeAttributes</code>	boolean	Read-write. True to add width and height attributes for images (default: <code>true</code> ).
<code>bannerOptions</code>	<a href="#">GalleryBannerOptions</a>	Read-write. The options related to banner settings.
<code>customColorOptions</code>	<a href="#">GalleryCustomColorOptions</a>	Read-write. The options related to custom color settings.
<code>emailAddress</code>	string	Read-write. The email address to show on the web page.
<code>imagesOptions</code>	<a href="#">GalleryImagesOptions</a>	Read-write. The options related to images settings.
<code>includeSubFolders</code>	boolean	Read-write. True to include all files found in sub folders of the input folder (default: <code>true</code> ).
<code>layoutStyle</code>	string	Read-write. The style to use for laying out the web page (default: <code>Centered Frame 1 - Basic</code> ).
<code>preserveAllMetadata</code>	boolean	Read-write. True to save metadata (default: <code>false</code> ).
<code>securityOptions</code>	<a href="#">GallerySecurityOptions</a>	Read-write. The options related to security settings.
<code>thumbnailOptions</code>	<a href="#">GalleryThumbnailOptions</a>	Read-write. The options related to thumbnail image settings.
<code>typename</code>	string	Read-only. The class name of the referenced <code>galleryOptions</code> object.
<code>useShortExtension</code>	boolean	Read-write. True to use the short web page extension <code>.htm</code> . If false, use the web page extension <code>.html</code> (default: <code>true</code> ).
<code>useUTF8Encoding</code>	boolean	Read-write. True to use UTF-8 encoding for the web page (default: <code>false</code> ).

# GallerySecurityOptions

Options for the [securityOptions](#) property of the [GalleryOptions](#) object.

**Tip:** You can preserve default values for many of these properties by setting the [GalleryOptions](#) property [preserveAllMetadata](#) to true; this is the same as choosing **File > Automate > Web Photo Gallery**, and then choosing **Preserve all metadata** in the Options area of the Web Photo Gallery dialog.

## Properties

Property	Value type	What it is
<b>content</b>	<a href="#">GallerySecurityType</a>	Read-write. The web photo gallery security content (default: <code>GallerySecurityType.NONE</code> ).
<b>font</b>	<a href="#">GalleryFontType</a>	Read-write. The web photo gallery security font (default: <code>GalleryFontType.ARIAL</code> ).
<b>fontSize</b>	number [1..72]	Read-write. The web photo gallery security font size (default: 3).
<b>opacity</b>	number	Read-write. The web page security opacity as a percent (default: 100).
<b>text</b>	string	Read-write. The web photo gallery security custom text.
<b>textColor</b>	<a href="#">GallerySecurityTextColorType</a>	Read-write. The web page security text color.
<b>textPosition</b>	<a href="#">GallerySecurityTextPositionType</a>	Read-write. The web photo gallery security text position (default: <code>GallerySecurityTextPositionType.CENTERED</code> ).
<b>textRotate</b>	<a href="#">GallerySecurityTextRotateType</a>	Read-write. The web photo gallery security text orientation to use (default: <code>GallerySecurityTextRotateType.ZERO</code> ).
<b>typename</b>	string	Read-only. The class name of the referenced <code>gallerySecurityOptions</code> object.

# GalleryThumbnailOptions

Options for the [thumbnailOptions](#) property of the [GalleryOptions](#) object.

**Tip:** You can preserve default values for many of these properties by setting the [GalleryOptions](#) property [preserveAllMetadata](#) to true; this is the same as choosing **File > Automate > Web Photo Gallery**, and then choosing **Preserve all metadata** in the Options area of the Web Photo Gallery dialog.

## Properties

Property	Value type	What it is
<b>border</b>	number [0..99]	Read-write. The amount of border pixels you want around your thumbnail images (default: 0).
<b>caption</b>	boolean	Read-write. True if there is a caption (default: false).
<b>columnCount</b>	number	Read-write. The number of columns on the page (default: 5).
<b>dimension</b>	number	Read-write. The web photo gallery thumbnail dimension in pixels (default: 75).
<b>font</b>	<a href="#">GalleryFontType</a>	Read-write. The web photo gallery font (default: <code>GalleryFontType.ARIAL</code> ).
<b>fontSize</b>	number [1..7]	Read-write. The font size for thumbnail images text (default: 3).
<b>includeCopyright</b>	boolean	Read-write. True to include copyright information for thumbnails (default: false).
<b>includeCredits</b>	boolean	Read-write. True to include credits for thumbnails (default: false).
<b>includeFilename</b>	boolean	Read-write. True to include file names for thumbnails (default: false).
<b>includeTitle</b>	boolean	Read-write. True to include titles for thumbnails (default: false).
<b>rowCount</b>	number	Read-write. The number of rows on the page (default: 3).
<b>size</b>	<a href="#">GalleryThumbSizeType</a>	Read-write. The thumbnail image size (default: <code>GalleryThumbSizeType.MEDIUM</code> ).
<b>typename</b>	string	Read-only. The class name of the referenced <code>GalleryThumbnailOptions</code> object.

# GIFSaveOptions

Options for saving a document in GIF format using the [Document.saveAs\(\)](#) method.

## Properties

Property	Value type	What it is
<b>colors</b>	number	Read-write. The number of palette colors.  Valid only when <a href="#">palette</a> = <code>Palette.LOCALADAPTIVE</code> , <code>LOCALPERCEPTUAL</code> , <code>LOCALSELECTIVE</code> , <code>MACOSPALETTE</code> , <code>UNIFORM</code> , <code>WEBPALETTE</code> ; or <code>WINDOWSPALETTE</code> .
<b>dither</b>	<a href="#">Dither</a>	Read-write. The dither type.
<b>ditherAmount</b>	number [1..100]	Read-write. The amount of dither (default: 75).  Valid only when <a href="#">dither</a> = <code>Dither.DIFFUSION</code> .
<b>forced</b>	<a href="#">ForcedColors</a>	Read-write. The type of colors to force into the color palette.
<b>interlaced</b>	boolean	Read-write. True if rows should be interlaced (default: <code>false</code> ).
<b>matte</b>	<a href="#">MatteType</a>	Read-write. The color to use to fill anti-aliased edges adjacent to transparent areas of the image (default: <code>MatteType.WHITE</code> ).  When <a href="#">transparency</a> = <code>false</code> , the matte color is applied to transparent areas.
<b>palette</b>	<a href="#">PaletteType</a>	Read-write. The type of palette to use (default: <code>Palette.LOCALSELECTIVE</code> ).
<b>preserveExactColors</b>	boolean	Read-write. True to protect colors in the image that contain entries in the color table from being dithered.  Valid only when <a href="#">dither</a> = <code>Dither.DIFFUSION</code> .
<b>transparency</b>	boolean	Read-write. True to preserve transparent areas of the image during conversion to GIF format.
<b>typename</b>	string	Read-only. The class name of the referenced <code>GIFSaveOptions</code> object.

# GrayColor

Defines a gray color, used in the [SolidColor](#) object.

See also [CMYKColor](#), [HSBColor](#), [LabColor](#), [NoColor](#), [RGBColor](#)

## Properties

Property	Value type	What it is
<b>gray</b>	number [0.0..100.0]	Read-write. The gray value (default: 0.0).
<b>typename</b>	string	Read-only. The class name of the referenced <code>grayColor</code> object.

# Guide

## Properties

Property	Value type	What it is
<b>direction</b>	<a href="#">Direction</a>	Read-write. Indicates whether the guide is vertical or horizontal.
<b>coordinate</b>	<a href="#">UnitValue</a>	Read-write. Location of the guide from origin of image.

## Guides

The collection of open [Guide](#) objects.

Access this list through the [Document.guides](#) collection property, which is available through the `activeDocument` object. For example, the following adds a new guide to the collection:

```
activeDocument.guides.add (Direction.HORIZONTAL,UnitValue(20,20))
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>guides</code> collection.
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>typename</b>	string	Read-only. The class name of the referenced <code>guides</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>add</b> (direction , coordinate)	<a href="#">Direction</a> <a href="#">UnitValue</a>	<a href="#">Guide</a>	Creates a new guide object and adds it to this collection.
<b>getByName</b> (name)	string	<a href="#">Guide</a>	Gets the first element in the <code>guides</code> collection with the provided name



# HistoryState

A version of the document stored automatically (and added to the [HistoryStates](#) collection), which preserves the document's state, each time the document is changed.

Access through [Document.historyStates](#) collection. You can access a state in the list by name. For example, this assigns a property value in the state object named "AddLayerMask":

```
var stateRef = app.activeDocument.historyState.getByName("AddLayerMask");  
stateRef.snapshot = true;
```

## Properties

Property	Value type	What it is
<b>name</b>	string	Read-only. The HistoryState object's name.
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>snapshot</b>	boolean	Read-only. True if the history state is a snapshot.
<b>typename</b>	string	Read-only. The class name of the referenced HistoryState object.

# HistoryStates

The collection of [Guide](#) objects in the document.

Access through [Document.historyStates](#) collection property. For example, this accesses one of the states in the collection by index:

```
myState = app.activeDocument.historyStates[7];
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>HistoryStates</code> collection.
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>typename</b>	string	Read-only. The class name of the referenced <code>HistoryStates</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>getByName</b> (name)	string	<a href="#">Guide</a>	Get the first element in the <code>HistoryStates</code> collection with the provided name.

# HSBColor

Defines an HSB color, used in the [SolidColor](#) object.

See also [CMYKColor](#), [GrayColor](#), [LabColor](#), [NoColor](#), [RGBColor](#)

## Properties

Property	Value type	What it is
<b>brightness</b>	number [0.0..100.0]	Read-write. The brightness value.
<b>hue</b>	number [0.0..360.0]	Read-write. The hue value.
<b>saturation</b>	number [0.0..100.0]	Read-write. The saturation value.
<b>typename</b>	string	Read-only. The class name of the referenced <code>HSBColor</code> object.

# IndexedConversionOptions

Options for converting an RGB image to an indexed color model using [Document.changeMode\(\)](#).

## Properties

Property	Value type	What it is
<b>colors</b>	number	Read-write. The number of palette colors.  Valid only when <a href="#">palette</a> = <code>Palette.LOCALADAPTIVE</code> , <code>LOCALPERCEPTUAL</code> , <code>LOCALSELECTIVE</code> , <code>MACOSPALETTE</code> , <code>UNIFORM</code> , <code>WEBPALETTE</code> , or <code>WINDOWSPALETTE</code> .
<b>dither</b>	<a href="#">Dither</a>	Read-write. The dither type.
<b>ditherAmount</b>	number [1..100]	Read-write. The amount of dither.  Valid only when <a href="#">dither</a> = <code>Dither.diffusion</code> .
<b>forced</b>	<a href="#">ForcedColors</a>	Read-write. The type of colors to force into the color palette.
<b>matte</b>	<a href="#">MatteType</a>	Read-write. The color to use to fill anti-aliased edges adjacent to transparent areas of the image (default: <code>MatteType.WHITE</code> ).  When <a href="#">transparency</a> = <code>false</code> , the matte color is applied to transparent areas.
<b>palette</b>	<a href="#">PaletteType</a>	Read-write. The palette type (default: <code>Palette.EXACT</code> ).
<b>preserveExactColors</b>	boolean	Read-write. True to protect colors in the image that contain entries in the color table from being dithered.  Valid only when <a href="#">dither</a> = <code>Dither.DIFFUSION</code> .
<b>transparency</b>	boolean	Read-write. True to preserve transparent areas of the image during conversion to GIF format.
<b>typename</b>	string	Read-only. The class name of the referenced <code>IndexedConversionOptions</code> object.

# JPEGSaveOptions

Options for saving a document in JPEG format using the [Document.saveAs\(\)](#) method.

## Properties

Property	Value type	What it is
<b>embedColorProfile</b>	boolean	Read-write. True to embed the color profile in the document.
<b>formatOptions</b>	<a href="#">FormatOptions</a>	Read-write. The download format to use (default: <code>FormatOptions.STANDARDBASELINE</code> ).
<b>matte</b>	<a href="#">MatteType</a>	<p>Read-write. The color to use to fill anti-aliased edges adjacent to transparent areas of the image (default: <code>MatteType.WHITE</code>).</p> <p>When transparency is turned off for an image, the matte color is applied to transparent areas.</p>
<b>quality</b>	number [0..12]	Read-write. The image quality setting to use; affects file size and compression (default: 3).
<b>scans</b>	number [3..5]	<p>Read-write. The number of scans to make to incrementally display the image on the page (default: 3).</p> <p>Valid only for when <a href="#">formatOptions</a> = <code>FormatOptions.PROGRESSIVE</code>.</p>
<b>typename</b>	string	Read-only. The class name of the referenced <code>JPEGSaveOptions</code> object.

# LabColor

Defines an LAB color, used in the [SolidColor](#) object.

See also [CMYKColor](#), [GrayColor](#), [HSBColor](#), [NoColor](#), [RGBColor](#)

## Properties

Property	Value type	What it is
<b>a</b>	number [-128.0..127.0]	Read-write. The a-value.
<b>b</b>	number [-128.0..127.0]	Read-write. The b-value.
<b>l</b>	number [0.0..100.0]	Read-write. The L-value.
<b>typename</b>	string	Read-only. The class name of the referenced <code>LabColor</code> object.

# LayerComp

A snapshot of a state of the layers in a document, which can be used to view different page layouts or compositions.

Access through [Document.layerComps](#) collection. You can access a layer comp by its name. For example, this sets the `comment` property value for a `LayerComp` object named `myLayerComp`:

```
var layercompRef = app.activeDocument.layerComps.getByName("myLayerComp");
layercompRef.comment = "View from shoreline";
```

## Properties

Property	Value type	What it is
<b>appearance</b>	boolean	Read-write. True to use layer appearance (layer styles) settings.
<b>comment</b>	string	Read-write. A description of the layer comp.
<b>name</b>	string	Read-write. The name of the layer comp.
<b>parent</b>	<a href="#">Document</a>	Read-write. The containing document.
<b>position</b>	boolean	Read-write. True to use layer position.
<b>selected</b>	boolean	Read-only. True if the layer comp is currently selected.
<b>typename</b>	string	Read-only. The class name of the referenced <code>layerComp</code> object.
<b>visibility</b>	boolean	Read-write. True to use layer visibility settings .

## Methods

Method	Parameter type	Returns	What it does
<b>apply</b> ( )			Applies the layer comp to the document.
<b>recapture</b> ( )			Recaptures the current layer state(s) for this layer comp.
<b>remove</b> ( )			Deletes the <code>layerComp</code> object.
<b>resetfromComp</b> ( )			Resets the layer comp state to the document state.

# LayerComps

The collection of [LayerComp](#) objects in the document.

Access through the [Document.layerComps](#) collection property. For example:

```
app.activeDocument.layerComps.add("myLayerComp", "View from Shoreline",  
    true, true, true);
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>layerComps</code> collection.
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>typename</b>	string	Read-only. The class name of the referenced <code>layerComps</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>add</b> (name, comment, appearance, position, visibility)	string string boolean boolean boolean	<a href="#">LayerComp</a>	Creates a new layer composition object and adds it to this collection.
<b>getByName</b> (name)	string	<a href="#">LayerComp</a>	Gets the first element in the collection with the provided name.
<b>removeAll</b> ( )			Removes all member objects from the <code>layerComps</code> collection.



# Layers

The collection of layer objects, including [ArtLayer](#) and [LayerSet](#) objects, in the document. Access through [Document.layers](#), or the [LayerSet.layers](#) collections properties.

For example, this uses the `length` property to count the number of `layer` objects in the active document, then displays the number on the screen:

```
var layerNum = app.activeDocument.layers.length
alert(layerNum)
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>layers</code> collection.
<b>parent</b>	<a href="#">Document</a> or <a href="#">LayerSet</a>	Read-only. The containing document or layer set.
<b>typename</b>	string	Read-only. The class name of the referenced <code>layers</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>getByName</b> (name)	string	Layer	Gets the first element in the <code>layers</code> collection with the provided name.
<b>removeAll</b> ( )			Removes all layers from the collection.

# LayerSet

A group of layer objects, which can include [ArtLayer](#) objects and other (nested) `LayerSet` objects. A single command can manipulate all layers in the set.

Access top-level layers sets in a document through the [Document.layerSets](#) collection. You can access a layer set by name. For example, the following sets the `allLocked` value for "myLayerSet":

```
var layerSetRef = app.activeDocument.layerSets.getByName("myLayerSet");
layerSetRef.allLocked = true
```

Access nested layer sets through the [LayerSet.layerSets](#) collection in the parent set. For example:

```
app.activeDocument.layerSets[0].layerSets[0];
```

## Properties

Property	Value type	What it is
<code>allLocked</code>	boolean	Read-write. True if the contents in the layers in this set are not editable.
<code>artLayers</code>	<a href="#">ArtLayers</a>	Read-only. The art layers in this layer set.
<code>blendMode</code>	<a href="#">BlendMode</a>	Read-write. The blend mode to use for the layer set.
<code>bounds</code>	array of <a href="#">UnitValue</a>	Read-only. The bounding rectangle of the layer set.
<code>enabledChannels</code>	array of <a href="#">Channel</a>	Read-write. The channels enabled for the layer set; must be a list of component channels. See <a href="#">Channel.kind</a> .
<code>layers</code>	<a href="#">Layers</a>	Read-only. The layers in this layer set.
<code>layerSets</code>	<a href="#">LayerSets</a>	Read-only. Nested layer sets contained within this layer set.
<code>linkedLayers</code>	array of <a href="#">ArtLayer</a> and/or <a href="#">LayerSet</a>	Read-only. The layers linked to this <code>layerSet</code> object.
<code>name</code>	string	Read-write. The name of this layer set.
<code>opacity</code>	number [0.0..100.0]	Read-write. The master opacity of the set.
<code>parent</code>	<a href="#">Document</a> or <a href="#">LayerSet</a>	Read-only. The containing document or layer set.
<code>typename</code>	string	Read-only. The class name of the referenced <code>LayerSet</code> object.
<code>visible</code>	boolean	Read-write. True if the set is visible.

## Methods

Method	Parameter type	Returns	What it does
<b>duplicate</b> ([relativeObject]  [, insertionLocation])	<a href="#">ArtLayer</a> or <a href="#">LayerSet</a> <a href="#">ElementPlacement</a>	<a href="#">LayerSet</a>	Creates a duplicate of the object.
<b>link</b> (with)	<a href="#">ArtLayer</a> or <a href="#">LayerSet</a>		Links the layer set with another layer.
<b>merge</b> ( )		<a href="#">ArtLayer</a>	Merges the layerset; returns a reference to the art layer created by this method.
<b>move</b> (relativeObject,  insertionLocation)	<a href="#">ArtLayer</a> or <a href="#">LayerSet</a> <a href="#">ElementPlacement</a>		Moves the object.
<b>remove</b> ( )			Deletes the object.
<b>resize</b> ([horizontal] [, vertical] [, anchor])	number number <a href="#">AnchorPosition</a>		Resizes all layers in the layer set to the specified dimensions (as a percentage of its current size) and places the layer set in the specified position.
<b>rotate</b> (angle [, anchor])	number <a href="#">AnchorPosition</a>		Rotates all layers in the layer set around the specified anchor point (default: <code>AnchorPosition.MIDDLECENTER</code> )
<b>translate</b> ([deltaX] [, deltaY])	<a href="#">UnitValue</a> <a href="#">UnitValue</a>		Moves the position relative to its current position.
<b>unlink</b> ( )			Unlinks the layer set.

## LayerSets

The collection of [LayerSet](#) objects in the document.

Access the top-level layer sets in a document through the [Document.layerSets](#) collection property. For example:

```
var layerSetRef = app.activeDocument.layerSets.add()
```

Access the nested layer sets through the [LayerSet.layerSets](#) collection property in the parent set. For example:

```
var layerSetRef = app.activeDocument.layerSets.getByName("myParentSet");  
var childSet = layerSetRef.layerSets.getByName("myChildSet");
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>LayerSets</code> collection.
<b>parent</b>	<a href="#">Document</a> or <a href="#">LayerSet</a>	Read-only. The containing document or layer set.
<b>typename</b>	string	Read-only. The class name of the referenced <code>layerSets</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>add</b> ( )		<a href="#">LayerSet</a>	Creates a new layer set object and adds it to the collection.
<b>getByName</b> (name)	string	<a href="#">LayerSet</a>	Gets the first element in the collection with the provided name.
<b>removeAll</b> ( )			Removes all member layer sets, and any layers or layer sets they contain, from the document.

## LayerSets sample script

The following script creates three layer sets, then nests a second layer set in each layer set, and then creates a text layer in each nested set that displays the text "Layer in *n* Set Inside *n* Set", where *n* represents the ordinal number of the set (first, second, or third).

**Note:** This script uses the ExtendScript `$` debugging object. For further details, see the *JavaScript Tools Guide*.

### LayerSets.jsx

```
$.level = 1  
  
//close all open documents  
while (app.documents.length) {
```

```
    app.activeDocument.close()
}

// create a working document
var docRef = app.documents.add()

// create an array to hold the layer sets
var myLayerSets = new Array()

// Create an array to hold the text
var textArray = Array("First", "Second", "Third")

//Create an indexer variable
var i = 0

// Create three layer sets at the top level
for (i = 0; i < 3; i++) {
    myLayerSets[i] = new Array()
    myLayerSets[i][0] = docRef.layerSets.add()
}

// Rearrange the layer sets with the first one on top, second next, etc.
myLayerSets[1][0].moveAfter(myLayerSets[0][0])
myLayerSets[2][0].moveAfter(myLayerSets[1][0])

// Create a layer set inside each layer set
for (i = 0; i < 3; i++) {
    myLayerSets[i][0].name = textArray[i] + " Set"
    myLayerSets[i][1] = myLayerSets[i][0].layerSets.add()
    myLayerSets[i][1].name = "Inside " + textArray[i] + " Set"
}

// Create an array to hold the layers
var myLayers = new Array()

// Create a text layer with a description inside each layer set
for (i = 0; i < 3; i++) {
    myLayers[i] = myLayerSets[i][1].artLayers.add()
    myLayers[i].kind = LayerKind.TEXT
    myLayers[i].textItem.contents = "Layer in " + textArray[i] + " Set Inside "
        + textArray[i] + " Set"
    myLayers[i].textItem.position = Array(app.activeDocument.width * i * 0.33,
        app.activeDocument.height * (i + 1) * 0.25)
    myLayers[i].textItem.size = 12
}
```

# MeasurementLog

The measurement log for the application. Access through the [Application.measurementLog](#) property.

**Note:** This feature is available in the Extended Version only.

## Methods

Method	Parameter type	Returns	What it does
<b>exportMeasurements</b> ([file] [, range] [, dataPoints])	<a href="#">File</a> <a href="#">MeasurementRange</a> array of string		Export measurement to a file.
<b>deleteMeasurements</b> ([range])	<a href="#">MeasurementRange</a>		Delete measurements from the log.

## MeasurementScale

The measurement scale for the document. Access through the [Document.measurementScale](#) property. For example:

```
app.activeDocument.measurementScale.pixelLength = 25
```

**Note:** This feature is available in the Extended Version only.

### Properties

Property	Value type	What it is
<code>pixelLength</code>	number	Read-write. The length in pixels this scale equates to.
<code>logicalLength</code>	number	Read-write. The logical length this scale equates to.
<code>logicalUnits</code>	string	Read-write. The logical units for this scale.

# NoColor

Represents a missing color object, used in the [SolidColor](#) object.

See also [CMYKColor](#), [GrayColor](#), [HSBColor](#), [LabColor](#), [RGBColor](#)

## Properties

Property	Value type	What it is
<code>typename</code>	string	Read-only. The class name of the referenced <code>noColor</code> object.



# Notifier

An event-handler object that tells a script to execute specified code when a specified event occurs. Notifiers must be enabled using the [Application.notifiersEnabled](#) property.

Access through the [Application.notifiers](#) collection.

**Note:** Events that occur within scripts do not generally trigger notifiers, because they occur inside a "play script" event.

## Properties

Property	Value type	What it is
<b>event</b>	string	Read-only. The event identifier, a four-character code or a unique string.  For a list of four-character codes, see <a href="#">Appendix A: Event ID Codes</a> .
<b>eventClass</b>	string	Read-only. The class identifier, a four-character code or a unique string.  When an event applies to multiple types of objects, use this property to distinguish which object this notifier applies to. For example, the Make event ("Mk ") can apply to documents ("Dcmn"), channels ("Chnl") and other objects.
<b>eventFile</b>	<a href="#">File</a>	Read-only. The path to the file to execute when the event occurs and activates the notifier.
<b>parent</b>	<a href="#">Application</a>	Read-only. The containing application.
<b>typename</b>	string	Read-only. The class name of the referenced object.

## Methods

Method	Parameter type	Returns	What it does
<b>remove</b> ( )			Deletes this object.  You can also remove a <code>Notifier</code> object from the Script Events Manager drop-down list by deleting the file named <code>Script Events Manager.xml</code> from the Photoshop preferences folder. See Adobe Photoshop CC 2015 help for more information.

## Notifiers

The collection of [Notifier](#) objects in the document. Access through the [Application.notifiers](#) collection property. For example:

```
var notRef = app.notifiers.add("OnClickGoButton", eventFile)
```

Notifiers must be enabled using the [Application.notifiersEnabled](#) property.

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the <code>notifiers</code> collection.
<b>parent</b>	<a href="#">Application</a>	Read-only. The <code>notifiers</code> object's container
<b>typename</b>	string	Read-only. The class name of the referenced <code>notifiers</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>add</b> (event, eventFile [, eventClass])	string <a href="#">File</a> string	<a href="#">Notifier</a>	<p>Creates a notifier object and adds it to this collection.</p> <p><code>event</code> defines the class ID of the event: use a 4-characters code or a unique string. See <a href="#">Appendix A: Event ID Codes</a>.</p> <p><code>eventFile</code> defines the script file that executes when the event occurs.</p> <p>When an event applies to multiple types of objects, use the <code>eventClass</code> (a 4-character ID or unique string) to distinguish which object this <code>Notifier</code> applies to. For example, the Make event ("Mk ") applies to documents ("Dcmn"), channels ("Chnl") and other objects.</p> <p><b>Tip:</b> When specifying an event or event calss wtih a 4-character ID code, omit the single quotes in your code.</p>
<b>removeAll</b> ( )			<p>Removes all member objects from the <code>notifiers</code> collection.</p> <p>You can also remove a <code>notifier</code> object from the Script Events Manager drop-down list by deleting the file named <code>Script Events Manager.xml</code> from the Photoshop preferences folder. See Adobe Photoshop CC 2015 help for more information.</p>

# PathItem

A path or drawing object, such as the outline of a shape or a straight or curved line, which contains sub paths that define its geometry.

Access through the collection in the [Document.pathItems](#) property. For example, this selects a named path item:

```
var currentPathItem = app.activeDocument.pathItems.getByNamed("myPath");
currentPathItem.select();
```

Create these objects by passing a set of [SubPathInfo](#) objects to the [PathItems.add\(\)](#) method. This method creates a [SubPathItem](#) object for each [SubPathInfo](#) object, and creates and returns a new [PathItem](#) object for the path represented by all of the subpaths.

## Properties

Property	Value type	What it is
<b>kind</b>	<a href="#">PathKind</a>	Read-write. The type.
<b>name</b>	string	Read-write. The name.
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>subPathItems</b>	<a href="#">SubPathItems</a>	Read-only. The contained sub-path objects.
<b>typename</b>	string	Read-only. The class name of the referenced <code>pathItem</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>deselect</b> ( )			Deselects this <code>pathItem</code> object.
<b>duplicate</b> (name)	string		Duplicates this <code>pathItem</code> object with the new name.
<b>fillPath</b> ( [fillColor] [, mode] [, opacity] [, preserveTransparency] [, feather] [, wholePath] [, antiAlias] )	<a href="#">SolidColor</a> <a href="#">ColorBlendMode</a> number [0..100] boolean number [0.0..250.0] boolean boolean		Fills the area enclosed by this path.  <code>opacity</code> is a percentage.  <code>feather</code> is in pixels.  If <code>wholePath</code> is true, all subpaths are used when doing the fill (default: true).

Method	Parameter type	Returns	What it does
<b>makeClippingPath</b> ([flatness])	number [0.2..100]		Makes this the clipping path for this document. <i>flatness</i> tells the PostScript printer how to approximate curves in the path.
<b>makeSelection</b> ([feather] [, antiAlias] [, operation])	number [0.0..250.0] boolean <a href="#">SelectionType</a>		Makes a <a href="#">Selection</a> object whose border is this path. <i>feather</i> is in pixels.
<b>remove</b> ( )			Deletes this object.
<b>select</b> ( )			Makes this the active or selected <a href="#">PathItem</a> object.
<b>strokePath</b> ([tool] [, simulatePressure])	<a href="#">ToolType</a> boolean		Strokes the path with the specified tool.

## PathItem sample script

The following creates a path in three segments: two diagonal lines that form a V, and a curved line above the V that makes it look like a 2D ice cream cone.

### Paths.jsx

```
// Save the current preferences
var startRulerUnits = app.preferences.rulerUnits
var startTypeUnits = app.preferences.typeUnits
var startDisplayDialogs = app.displayDialogs

// Set Adobe Photoshop CC 2015 to use pixels and display no dialogs
app.preferences.rulerUnits = Units.PIXELS
app.preferences.typeUnits = TypeUnits.PIXELS
app.displayDialogs = DialogModes.NO

// first close all the open documents
while (app.documents.length) {
    app.activeDocument.close()
}

// create a document to work with
var docRef = app.documents.add(5000, 7000, 72, "Simple Line")

//line 1--it's a straight line so the coordinates for anchor, left, and right
//for each point have the same coordinates
var lineArray = new Array()
    lineArray[0] = new PathPointInfo
    lineArray[0].kind = PointKind.CORNERPOINT
    lineArray[0].anchor = Array(100, 100)
```

```
lineArray[0].leftDirection = lineArray[0].anchor
lineArray[0].rightDirection = lineArray[0].anchor

lineArray[1] = new PathPointInfo
lineArray[1].kind = PointKind.CORNERPOINT
lineArray[1].anchor = Array(150, 200)
lineArray[1].leftDirection = lineArray[1].anchor
lineArray[1].rightDirection = lineArray[1].anchor

var lineSubPathArray = new Array()
lineSubPathArray[0] = new SubPathInfo()
lineSubPathArray[0].operation = ShapeOperation.SHAPEXOR
lineSubPathArray[0].closed = false
lineSubPathArray[0].entireSubPath = lineArray

// line 2
var lineArray2 = new Array()
lineArray2[0] = new PathPointInfo
lineArray2[0].kind = PointKind.CORNERPOINT
lineArray2[0].anchor = Array(150, 200)
lineArray2[0].leftDirection = lineArray2[0].anchor
lineArray2[0].rightDirection = lineArray2[0].anchor

lineArray2[1] = new PathPointInfo
lineArray2[1].kind = PointKind.CORNERPOINT
lineArray2[1].anchor = Array(200, 100)
lineArray2[1].leftDirection = lineArray2[1].anchor
lineArray2[1].rightDirection = lineArray2[1].anchor

lineSubPathArray[1] = new SubPathInfo()
lineSubPathArray[1].operation = ShapeOperation.SHAPEXOR
lineSubPathArray[1].closed = false
lineSubPathArray[1].entireSubPath = lineArray2

//ice cream curve
//it's a curved line, so there are 3 points, not 2
//coordinates for the middle point (lineArray3[1]) are different.
//The left direction is positioned "above" the anchor on the screen.
//The right direction is positioned "below" the anchor
//You can change the coordinates for these points to see
//how the curve works...
var lineArray3 = new Array()
lineArray3[0] = new PathPointInfo
lineArray3[0].kind = PointKind.CORNERPOINT
lineArray3[0].anchor = Array(200, 100)
lineArray3[0].leftDirection = lineArray3[0].anchor
lineArray3[0].rightDirection = lineArray3[0].anchor

lineArray3[1] = new PathPointInfo
lineArray3[1].kind = PointKind.CORNERPOINT
lineArray3[1].anchor = Array(150, 50)
lineArray3[1].leftDirection = Array(100, 50)
lineArray3[1].rightDirection = Array(200, 50)

lineArray3[2] = new PathPointInfo
lineArray3[2].kind = PointKind.CORNERPOINT
lineArray3[2].anchor = Array(100, 100)
lineArray3[2].leftDirection = lineArray3[2].anchor
lineArray3[2].rightDirection = lineArray3[2].anchor
```

```
    lineSubPathArray[2] = new SubPathInfo()
    lineSubPathArray[2].operation = ShapeOperation.SHAPEXOR
    lineSubPathArray[2].closed = false
    lineSubPathArray[2].entireSubPath = lineArray3

//create the path item
var myPathItem = docRef.pathItems.add("A Line", lineSubPathArray)

// stroke it so we can see something
myPathItem.strokePath(ToolType.BRUSH)

// Reset the application preferences
preferences.rulerUnits = startRulerUnits
preferences.typeUnits = startTypeUnits
displayDialogs = startDisplayDialogs
```

# PathItems

The collection of [PathItem](#) objects in a document.

Access through the [Document.pathItems](#) collection property. For example, this creates a new path item using a previously-defined set of subpaths:

```
app.activeDocument.pathItems.add("myPath", lineSubPathInfoArray);
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of <code>pathItem</code> objects in the <code>pathItems</code> collection.
<b>parent</b>	<a href="#">Document</a>	Read-only. The <code>pathItems</code> object's container.
<b>typename</b>	string	Read-only. The class name of the referenced <code>pathItems</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>add</b> (name, entirePath)	string array of <a href="#">SubPathInfo</a>	<a href="#">PathItem</a>	Creates a new path item object and adds it to this collection.  A new <a href="#">SubPathItem</a> object is created for each <code>SubPathInfo</code> object provided in <code>entirePath</code> , and those <code>SubPathItem</code> objects are added to the <a href="#">subPathItems</a> collection of the returned <a href="#">PathItem</a> .
<b>getByName</b> (name)	string	<a href="#">PathItem</a>	Get the first element in the <code>pathItems</code> collection with the provided name.
<b>removeAll</b> ( )			Removes all <code>pathItem</code> objects from the <code>pathItems</code> collection.

## PathPoint

Represents the anchor and control-handle endpoints for a path segment. Each point (the anchor point, left-direction point, and right-direction point) is an array containing X and Y position coordinates.

- Use the `PathPoint` object to retrieve information about the points that describe existing path segments. The properties are read-only. Access [PathPoint](#) objects through the [SubPathItem.pathPoints](#) property.
- Use [PathPointInfo](#) with [PathItems.add\(\)](#) to create path points. The properties are writeable.

For paths that are straight segments (not curved), the coordinates of all three points are the same. For curved segments, the the coordinates are different. The difference between the anchor point and the left or right direction points determines the arc of the curve. You use the left direction point to bend the curve "outward" or make it convex; you use the right direction point to bend the curve "inward" or make it concave.

### Properties

Property	Value type	What it is
<b>anchor</b>	array of number	Read-only. The X and Y coordinates of the anchor point of the curve.
<b>kind</b>	<a href="#">PointKind</a>	Read-only. The role (corner or smooth) this point plays in the containing path segment.
<b>leftDirection</b>	array of number	Read-only. The location of the left-direction endpoint ('in' position).
<b>parent</b>	<a href="#">SubPathItem</a>	Read-only. The containing subpath object.
<b>rightDirection</b>	array of number	Read-only. The location of the right-direction endpoint ('out' position).
<b>typename</b>	string	Read-only. The class name of the referenced <code>PathPoint</code> object.



## PathPointInfo

Used to create a [PathPoint](#), which represents the anchor and control-handle endpoints for a path segment. Each point (the anchor point, left-direction point, and right-direction point) is an array containing X and Y position coordinates.

- Use the JavaScript `new` operator to create these objects, and store them in the [SubPathInfo.entireSubPath](#) property before using that object to create a path item with [PathItems.add\(\)](#).  
See the [PathPointInfo sample script](#) below.
- The resulting [SubPathItem](#) objects contain the resulting [PathPoint](#) objects. Use the [PathPoint](#) object to retrieve information about the points that describe existing path segments. The properties are read-only.

For paths that are straight segments (not curved), the coordinates of all three points are the same. For curved segments, the the coordinates are different. The difference between the anchor point and the left or right direction points determines the arc of the curve. You use the left direction point to bend the curve "outward" or make it convex; you use the right direction point to bend the curve "inward" or make it concave.

## Properties

Property	Value type	What it is
<b>anchor</b>	array of number	Read-write. The X and Y coordinates of the anchor point of the curve.
<b>kind</b>	<a href="#">PointKind</a>	Read-write. The role (corner or smooth) this point plays in the containing path segment.
<b>leftDirection</b>	array of number	Read-write. The location of the left-direction endpoint ('in' position).
<b>rightDirection</b>	array of number	Read-write. The location of the right-direction endpoint ('out' position).
<b>typename</b>	string	Read-only. The class name of the referenced <code>PathPointInfo</code> object.

## PathPointInfo sample script

```
function drawLine(doc, start, stop) {  
  
    var startPoint = new PathPointInfo();  
    startPoint.anchor = start;  
    startPoint.leftDirection = start;  
    startPoint.rightDirection = start;  
    startPoint.kind = PointKind.CORNERPOINT;  
  
    var stopPoint = new PathPointInfo();  
    stopPoint.anchor = stop;  
    stopPoint.leftDirection = stop;  
    stopPoint.rightDirection = stop;  
    stopPoint.kind = PointKind.CORNERPOINT;  
}
```

```
var spi = new SubPathInfo();
spi.closed = false;
spi.operation = ShapeOperation.SHAPEXOR;
spi.entireSubPath = [startPoint, stopPoint];

var line = doc.pathItems.add("Line", [spi]);
line.strokePath(ToolType.PENCIL);
line.remove();
};

drawLine(app.activeDocument, [100,100], [200,200]);
```

# PathPoints

A collection of [PathPoint](#) objects that define a subpath, kept in the [SubPathItem.pathPoints](#) property.

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the collection.
<b>parent</b>	<a href="#">SubPathItem</a>	Read-only. The containing subpath object.
<b>typename</b>	string	Read-only. The class name of the referenced <code>PathPoints</code> object.

# PDFOpenOptions

Options for opening a document in generic Adobe PDF format using the [Application.open\(\)](#) method.

## Properties

Property	Value type	What it is
<b>antiAlias</b>	boolean	Read-write. True to use antialias.
<b>bitsPerChannel</b>	<a href="#">BitsPerChannelType</a>	Read-write. The number of bits per channel.
<b>constrainProportions</b>	boolean	DEPRECATED for Adobe Photoshop CC 2015.
<b>cropPage</b>	<a href="#">CropToType</a>	Read-write. The method of cropping to use.
<b>height</b>	<a href="#">UnitValue</a>	DEPRECATED for Adobe Photoshop CC 2015.
<b>mode</b>	<a href="#">OpenDocumentMode</a>	Read-write. The color model to use.
<b>name</b>	string	Read-write. The name of the object.
<b>object</b>	number	Read-write. The number of 3d objects to open.
<b>page</b>	number	Read-write. The page or image to which to open the document, depending on the value of <a href="#">usePageNumber</a> .
<b>resolution</b>	number	Read-write. The resolution of the document (in pixels per inch).
<b>suppressWarnings</b>	boolean	Read-write. True to suppress warnings when opening the document.
<b>typename</b>	string	Read-only. The class name of the referenced <code>PDFOpenOptions</code> object.
<b>use3DObjectNumber</b>	boolean	Read-write. If true, the 3d property refers to using 3d object; if false, then <code>usePageNumber</code> is used.
<b>usePageNumber</b>	boolean	Read-write. When true, the <a href="#">page</a> property refers to a page number; when false, it refers to an image number.
<b>width</b>	<a href="#">UnitValue</a>	DEPRECATED for Adobe Photoshop CC 2015.

# PDFSaveOptions

Options for saving a document in Adobe PDF format using the [Document.saveAs\(\)](#) method.

## Properties

Property	Value type	What it is
<b>alphaChannels</b>	boolean	Read-write. True to save the alpha channels with the file.
<b>annotations</b>	boolean	Read-write. True to save comments with the file.
<b>colorConversion</b>	boolean	Read-write. True to convert the color profile to a destination profile.
<b>convertToEightBit</b>	boolean	Read-write. True to convert a 16-bit image to 8-bit for better compatibility with other applications.
<b>description</b>	string	Read-write. Description of the save options to use.
<b>destinationProfile</b>	string	Read-write. Description of the final RGB or CMYK output device, such as a monitor or a press standard.
<b>downgradeColorProfile</b>	boolean	DEPRECATED for Adobe Photoshop CC 2015.
<b>downSample</b>	<a href="#">PDFResample</a>	Read-write. The down sample method to use.
<b>downSampleSize</b>	number	Read-write. The size to downsample images if they exceed the limit in pixels per inch.
<b>downSampleSizeLimit</b>	number	Read-write. Limits downsampling or subsampling to images that exceed this value in pixels per inch.
<b>embedColorProfile</b>	boolean	Read-write. True to embed the color profile in the document.
<b>embedFonts</b>	boolean	DEPRECATED for Adobe Photoshop CC 2015.
<b>embedThumbnail</b>	boolean	Read-write. True to include a small preview image in Adobe PDF files.
<b>encoding</b>	<a href="#">PDFEncoding</a>	Read-write. The type of compression to use (default: <code>PDFEncoding.PDFZIP</code> ).
<b>interpolation</b>	boolean	DEPRECATED for Adobe Photoshop CC 2015.

Property	Value type	What it is (Continued)
<b>jpegQuality</b>	number [0..12]	Read-write. The quality of the produced image, which is inversely proportionate to the compression amount.  Valid only when <a href="#">encoding</a> = <code>PDFEncoding.JPEG</code> .
<b>layers</b>	boolean	Read-write. True to save the document's layers.
<b>optimizeForWeb</b>	boolean	Read-write. True to improve performance of PDF files on Web servers.
<b>outputCondition</b>	string	Read-write. An optional comment field for inserting descriptions of the output condition. The text is stored in the PDF/X file.
<b>outputConditionID</b>	string	Read-write. Identifier for the output condition.
<b>PDFCompatibility</b>	<a href="#">PDFCompatibility</a>	Read-write. The PDF version to make the document compatible with.
<b>PDFStandard</b>	<a href="#">PDFStandard</a>	Read-write. The PDF standard to make the document compatible with.
<b>preserveEditing</b>	boolean	Read-write. True to reopen the PDF in Adobe Photoshop CC 2015 with native Photoshop data intact.
<b>presetFile</b>	string	Read-write. The preset file to use for settings.  <b>Note:</b> This option overrides other settings.
<b>profileInclusionPolicy</b>	boolean	Read-write. True to show which profiles to include.
<b>registryName</b>	string	Read-write. URL where the output condition is registered.
<b>spotColors</b>	boolean	Read-write. True to save spot colors.
<b>tileSize</b>	number	Read-write. Compression option.  Valid only when <a href="#">encoding</a> = <code>PDFEncoding.JPEG2000</code> .
<b>transparency</b>	boolean	DEPRECATED for Adobe Photoshop CC 2015.
<b>typename</b>	string	Read-only. The class name of the referenced <code>PDFSaveOptions</code> object.
<b>useOutlines</b>	boolean	DEPRECATED for Adobe Photoshop CC 2015.

Property	Value type	What it is (Continued)
<b>vectorData</b>	boolean	DEPRECATED for Adobe Photoshop CC 2015.
<b>view</b>	boolean	Read-write. True to open the saved PDF in Adobe Acrobat.

## PhotoCDOpenOptions

**DEPRECATED** in Adobe Photoshop CC 2015. Kodak PhotoCD is now found in the Goodies folder on the Adobe Photoshop CC 2015 Install DVD.

Options for opening a document in Kodak Photo CD (PCD) format (including high-resolution files from Pro Photo CD discs) using the [Application.open\(\)](#) method.

### Properties

Property	Value type	What it is
<b>colorProfileName</b>	string	Read-write. The profile to use when reading the image.
<b>colorSpace</b>	<a href="#">PhotoCDColorSpace</a>	Read-write. The colorspace for the image.
<b>orientation</b>	<a href="#">Orientation</a>	Read-write. The image orientation.
<b>pixelSize</b>	<a href="#">PhotoCDSIZE</a>	Read-write. The image dimensions.
<b>resolution</b>	number	Read-write. The image resolution (in pixels per inch).
<b>typename</b>	string	Read-only. The class name of the referenced <code>photoCDOpenOptions</code> object.



# PhotoshopSaveOptions

Options for saving a document in PSD format using the [Document.saveAs\(\)](#) method.

## Properties

Property	Value type	What it is
<b>alphaChannels</b>	boolean	Read-write. True to save the alpha channels.
<b>annotations</b>	boolean	Read-write. True to save the annotations.
<b>embedColorProfile</b>	boolean	Read-write. True to embed the color profile in the document.
<b>layers</b>	boolean	Read-write. True to preserve the layers.
<b>spotColors</b>	boolean	Read-write. True to save the spot colors.
<b>typename</b>	string	Read-only. The class name of the referenced <code>photoshopSaveOptions</code> object.

# PICTFileSaveOptions

Options for saving a document in PICT format using the [Document.saveAs\(\)](#) method.

## Properties

Property	Value type	What it is
<b>alphaChannels</b>	boolean	Read-write. True to save the alpha channels.
<b>compression</b>	<a href="#">PICTCompression</a>	Read-write. The type of compression to use (default: <code>PICTCompression.NONE</code> ).
<b>embedColorProfile</b>	boolean	Read-write. True to embed the color profile in the document.
<b>resolution</b>	<a href="#">PICTBitsPerPixels</a>	Read-write. The number of bits per pixel.
<b>typename</b>	string	Read-only. The class name of the referenced <code>PICTFileSaveOptions</code> object.

## PICTResourceSaveOptions

Options for saving a document as a PICT Resource file using the [Document.saveAs\(\)](#) method.

### Properties

Property	Value type	What it is
<b>alphaChannels</b>	boolean	Read-write. True to save the alpha channels.
<b>compression</b>	<a href="#">PICTCompression</a>	Read-write. The type of compression to use (default: <code>PICTCompression.NONE</code> ).
<b>embedColorProfile</b>	boolean	Read-write. True to embed the color profile in the document.
<b>name</b>	string	Read-write. The name of the PICT resource.
<b>resolution</b>	<a href="#">PICTBitsPerPixels</a>	Read-write. The number of bits per pixel.
<b>resourceID</b>	number	Read-write. The ID of the PICT resource (default: 128).
<b>typename</b>	string	Read-only. The class name of the referenced <code>PICTResourceSaveOptions</code> object.

# PicturePackageOptions

Options for a picture package created with [Application.makePicturePackage\(\)](#).

## Properties

Property	Value type	What it is
<b>content</b>	<a href="#">PicturePackageTextType</a>	Read-write. The content information (default: <code>PicturePackageTextType.NONE</code> ).
<b>flatten</b>	boolean	Read-write. True if all layers in the final document are flattened (default: <code>true</code> ).
<b>font</b>	<a href="#">GalleryFontType</a>	Read-write. The font used for security text (default: <code>GalleryFontType.ARIAL</code> ).
<b>fontSize</b>	number	Read-write. The font size used for security text (default: <code>12</code> ).
<b>layout</b>	string	Read-write. The layout to use to generate the picture package (default: <code>"(2) 5x7"</code> ).
<b>mode</b>	<a href="#">NewDocumentMode</a>	Read-write. Read-write. The color profile to use as the document mode (default: <code>NewDocumentMode.RGB</code> ).
<b>opacity</b>	number	Read-write. The web page security opacity as a percent (default: <code>100</code> ).
<b>resolution</b>	number	Read-write. The resolution of the document in pixels per inch (default: <code>72.0</code> ).
<b>text</b>	string	Read-write. The picture package custom text. Valid only when <a href="#">content</a> = <code>PicturePackageType.USER</code> .
<b>textColor</b>	<a href="#">RGBColor</a>	Read-write. The color to use for security text.
<b>textPosition</b>	<a href="#">GallerySecurityTextPositionType</a>	Read-write. The security text position (default: <code>GallerySecurityTextPositionType.CENTERED</code> ).
<b>textRotate</b>	<a href="#">GallerySecurityTextRotateType</a>	Read-write. The orientation to use for security text (default: <code>GallerySecurityTextRotateType.ZERO</code> ).
<b>typename</b>	string	Read-only. The class name of the referenced <code>PicturePackageOptions</code> object.

## PixarSaveOptions

Options for saving a document in Pixar format using the [Document.saveAs\(\)](#) method.

### Properties

Property	Value type	What it is
<b>alphaChannels</b>	boolean	Read-write. True to save the alpha channels.
<b>typename</b>	string	Read-only. The class name of the referenced <code>PixarSaveOptions</code> object.

# PNGSaveOptions

Options for saving a document in PNG format using the [Document.saveAs\(\)](#) method.

## Properties

Property	Value type	What it is
<b>compression</b>	number [0..9]	Read-write. The compression value (default: 0).
<b>interlaced</b>	boolean	Read-write. True to interlace rows (default: <i>false</i> ).
<b>typename</b>	string	Read-only. The class name of the referenced <code>PNGSaveOptions</code> object.

## Preferences

Represents application preferences for Photoshop. Access this object through the [Application.preferences](#) property. For example:

```
app.preferences.rulerUnits = Units.PIXELS
app.preferences.typeUnits = TypeUnits.PIXELS
```

Setting values in this object is equivalent to selecting **Edit > Preferences** (in Windows) or **Photoshop > Preferences** (in Mac OS) in the Adobe Photoshop CC 2015 application. For explanations of individual settings, see Adobe Photoshop CC 2015 Help.

## Properties

Property	Value type	What it is
<code>additionalPluginFolder</code>	<a href="#">File</a>	Read-write. The path to an additional plug-in folder. Valid only when <a href="#">useAdditionalPluginFolder</a> = <code>true</code> .
<code>appendExtension</code>	<a href="#">SaveBehavior</a>	Read-write. The preferred policy for writing file extensions in Windows.
<code>askBeforeSavingLayeredTIFF</code>	boolean	Read-write. True to ask the user to verify layer preservation options when saving a file in TIFF format.
<code>autoUpdateOpenDocuments</code>	boolean	Read-write. True to automatically update open documents.
<code>beepWhenDone</code>	boolean	Read-write. True to beep when a process finishes.
<code>colorChannelsInColor</code>	boolean	Read-write. True to display component channels in the Channels palette in color.
<code>colorPicker</code>	<a href="#">ColorPicker</a>	Read-write. The preferred color selection tool.
<code>columnGutter</code>	number [0.1..600.0]	Read-write. The width of the column gutters (in points).
<code>columnWidth</code>	number [0.1..600.0]	Read-write. Column width (in points)
<code>createFirstSnapshot</code>	boolean	Read-write. True to automatically make the first snapshot when a new document is created.
<code>dynamicColorSliders</code>	boolean	Read-write. True if dynamic color sliders appear in the Color palette.
<code>editLogItems</code>	<a href="#">EditLogItemsType</a>	Read-write. The preferred level of detail in the history log. Valid only when <a href="#">useHistoryLog</a> = <code>true</code> .

Property	Value type	What it is (Continued)
<b>exportClipboard</b>	boolean	Read-write. True to retain Adobe Photoshop CC 2015 contents on the clipboard after you exit the application.
<b>fontPreviewSize</b>	<a href="#">FontPreviewType</a>	Read-write. The preferred type size to use for font previews in the type tool font menus.
<b>fullSizePreview</b>	boolean	Read-write. True to show image preview as a full size image, false to show thumbnail (in Mac OS only).
<b>gamutWarningOpacity</b>	number [0..100]	Read-write. Opacity value as a percentage.
<b>gridSize</b>	<a href="#">GridSize</a>	Read-write. The preferred size to use for squares in the grid.
<b>gridStyle</b>	<a href="#">GridLineStyle</a>	Read-write. The preferred formatting style for non-printing grid lines.
<b>gridSubDivisions</b>	number [1..100]	Read-write. Number of grid subdivisions.
<b>guideStyle</b>	<a href="#">GuideLineStyle</a>	Read-write. The preferred formatting style for non-printing guide lines.
<b>iconPreview</b>	boolean	Read-write. True to use icon previews (in Mac OS only).
<b>imageCacheLevels</b>	number [1..8]	Read-write. The number of images to hold in the cache.
<b>imagePreviews</b>	<a href="#">SaveBehavior</a>	Read-write. The preferred policy for writing image previews in Windows.
<b>interpolation</b>	<a href="#">ResampleMethod</a>	Read-write. The method to use to assign color values to any new pixels created when an image is resampled or resized.
<b>keyboardZoomResizesWindows</b>	boolean	Read-write. True to automatically resize the window when zooming in or out using keyboard shortcuts.
<b>macOSThumbnail</b>	boolean	Read-write. True to create a thumbnail when saving the image (in Mac OS only).
<b>maximizeCompatibility</b>	<a href="#">QueryStateType</a>	Read-write. The preferred policy for checking whether to maximize compatibility when opening PSD files.
<b>maxRAMuse</b>	number [5..100]	Read-write. The maximum percentage of available RAM used by Adobe Photoshop CC 2015 (5 - 100).
<b>nonLinearHistory</b>	boolean	Read-write. True to allow non-linear history.



Property	Value type	What it is (Continued)
<b>numberOfHistoryStates</b>	number	Read-write. The number of history states to preserve.
<b>otherCursors</b>	<a href="#">OtherPaintingCursors</a>	Read-write. The preferred type of pointer to use with certain tools.
<b>paintingCursors</b>	<a href="#">PaintingCursors</a>	Read-write. The preferred type of pointer to use with certain tools.
<b>parent</b>	<a href="#">Application</a>	Read-write. The containing application.
<b>pixelDoubling</b>	boolean	Read-write. True to halve the resolution (double the size of pixels) to make previews display more quickly.
<b>pointSize</b>	<a href="#">PointType</a>	Read-write. The point/pica size.
<b>recentFileListLength</b>	number [0..30]	Read-write. The number of items in the recent file list.
<b>rulerUnits</b>	<a href="#">Units</a>	Read-write. The unit the scripting system will use when receiving and returning values.
<b>saveLogItems</b>	<a href="#">SaveLogItemsType</a>	Read-write. The preferred location of history log data when saving the history items.
<b>saveLogItemsFile</b>	<a href="#">File</a>	Read-write. The path to the history log file, when the preferred location is a file.
<b>savePaletteLocations</b>	boolean	Read-write. True to make new palette locations the default location.
<b>showAsianTextOptions</b>	boolean	Read-write. True to display Asian text options in the Paragraph palette.
<b>showEnglishFontNames</b>	boolean	Read-write. True to list Asian font names in English.
<b>showSliceNumber</b>	boolean	Read-write. True to display slice numbers in the document window when using the Slice tool.
<b>showToolTips</b>	boolean	Read-write. True to show pop up definitions on mouse over.
<b>smartQuotes</b>	boolean	Read-write. True to use curly, false to use straight quote marks.
<b>textFontSize</b>	<a href="#">FontSize</a>	Read-write. Size of the small font used in panels and dialogs.
<b>typename</b>	string	Read-only. The class name of the referenced <code>preferences</code> object.

Property	Value type	What it is (Continued)
<b>typeUnits</b>	<a href="#">TypeUnits</a>	Read-write. The preferred unit for text character measurements.
<b>useAdditionalPluginFolder</b>	boolean	Read-write. True to use an additional folder for compatible plug-ins stored with a different application.
<b>useHistoryLog</b>	boolean	Read-write. True to create a log file for history states.
<b>useLowerCaseExtension</b>	boolean	Read-write. True to use lowercase for file extensions.
<b>useShiftKeyForToolSwitch</b>	boolean	Read-write. True to enable cycling through a set of hidden tools.
<b>useVideoAlpha</b>	boolean	Read-write. True to enable Adobe Photoshop CC 2015 to send transparency information to your computer's video board. (Requires hardware support.)
<b>windowsThumbnail</b>	boolean	Read-write. True to create a thumbnail when saving the image in Windows. (Requires hardware support.)

# PresentationOptions

Options for Adobe PDF presentations created using [Application.makePDFPresentation\(\)](#).

## Properties

Property	Value type	What it is
<b>autoAdvance</b>	boolean	Read-write. True to auto advance images when viewing the presentation (default: <code>true</code> ). Valid only when <a href="#">presentation</a> = <code>true</code> .
<b>includeFilename</b>	boolean	Read-write. True to include the file name for the image (default: <code>false</code> ).
<b>interval</b>	number [1..60]	Read-write. The time in seconds before the view is auto advanced (default: 5). Valid only when <a href="#">autoAdvance</a> = <code>true</code> .
<b>loop</b>	boolean	Read-write. True to begin the presentation again after the last page (default: <code>false</code> ). Valid only when <a href="#">autoAdvance</a> = <code>true</code> .
<b>magnification</b>	<a href="#">MagnificationType</a>	Read-write. The magnification type to use when viewing the image.
<b>PDFFileOptions</b>	<a href="#">PDFSaveOptions</a>	Read-write. Options to use when creating the PDF file.
<b>presentation</b>	boolean	Read-write. True if the output will be a presentation (default: <code>false</code> ); when <code>false</code> , the output is a Multi-Page document.
<b>transition</b>	<a href="#">TransitionType</a>	Read-write. The method for transition from one image to the next (default: <code>TransitionType.NONE</code> ). Valid only when <a href="#">autoAdvance</a> = <code>true</code> .
<b>typename</b>	string	Read-only. The class name of the referenced <code>PresentationOptions</code> object.

# RawFormatOpenOptions

Options for opening a document in RAW format using the [Application.open\(\)](#) method.

## Properties

Property	Value type	What it is
<b>bitsPerChannel</b>	number	Read-write. The number of bits for each channel.  The only valid values are <code>BitsPerChannelType.EIGHT</code> or <code>BitsPerChannelType.SIXTEEN</code> .
<b>byteOrder</b>	<a href="#">ByteOrder</a>	Read-write. The order in which multibyte values are read.  Valid only when <a href="#">bitsPerChannel</a> = <code>BitsPerChannelType.SIXTEEN</code> .
<b>channelNumber</b>	number [1..56]	Read-write. The number of channels in the image. The value of cannot exceed the number of channels in the image.  When <a href="#">bitsPerChannel</a> = <code>BitsPerChannelType.SIXTEEN</code> , the only valid values are 1, 3, or 4.
<b>headerSize</b>	number [0..1919999]	Read-write. The number of bytes of information that will appear in the file before actual image information begins; that is, the number of zeroes inserted at the beginning of the file as placeholders.
<b>height</b>	number	Read-write. The height of the image (in pixels).
<b>interleaveChannels</b>	boolean	Read-write. True to store color values sequentially.
<b>retainHeader</b>	boolean	Read-write. True to retain the header when saving.  Valid only when <a href="#">headerSize</a> is 1 or greater.
<b>typename</b>	string	Read-only. The class name of the referenced <code>RawFormatOpenOptions</code> object.
<b>width</b>	number	Read-write. The image width in pixels.

## RawSaveOptions

Options for saving a document in RAW format using the [Document.saveAs\(\)](#) method.

### Properties

Property	Value type	What it is
<b>alphaChannels</b>	boolean	Read-write. True if alpha channels should be saved.
<b>spotColors</b>	boolean	Read-write. True if the spot colors should be saved.
<b>typename</b>	string	Read-only. The class name of the referenced <code>RawSaveOptions</code> object.

## RGBColor

Defines an RGB color, used in the [SolidColor](#) object.

See also [CMYKColor](#), [GrayColor](#), [HSBColor](#), [LabColor](#), [NoColor](#).

### Properties

Property	Value type	What it is
<b>blue</b>	number [0..255]	Read-write. The blue color value (default: 255).
<b>green</b>	number [0..255]	Read-write. The green color value (default: 255)
<b>hexValue</b>	string	Read-write. The hexadecimal representation of the color.
<b>red</b>	number [0..255]	Read-write. The red color value (default: 255)
<b>typename</b>	string	Read-only. The class name of the referenced <code>RGBColor</code> object.

## Selection

The selected area of a document or layer. Access through the [Document.selection](#) property. For example:

```
app.activeDocument.selection.fill(app.foregroundColor)
```

Many of the properties and methods use the [UnitValue](#) type, which combines measurement values with the measurement unit. For information about this type, see the *JavaScript Tools Guide*.

## Properties

Property	Value type	What it is
<b>bounds</b>	array of <a href="#">UnitValue</a>	Read-only. The bounding rectangle of the entire selection.
<b>parent</b>	<a href="#">Document</a>	Read-only. The object's container.
<b>solid</b>	boolean	Read-only. True if the bounding rectangle is a solid.
<b>typename</b>	string	Read-only. The class name of the referenced <code>selection</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>clear</b> ( )			Clears the selection and does not copy it to the clipboard.
<b>contract</b> (by)	<a href="#">UnitValue</a>		Contracts (reduces) the selection by the specified amount.
<b>copy</b> ( [merge] )	boolean		Copies the selection to the clipboard. When the optional argument is used and set to <code>true</code> , a merged copy is performed (all visible layers in the selection are copied).
<b>cut</b> ( )			Clears the current selection and copies it to the clipboard.
<b>deselect</b> ( )			Deselects the current selection.
<b>expand</b> (by)	<a href="#">UnitValue</a>		Expands the selection by the specified amount.
<b>feather</b> (by)	<a href="#">UnitValue</a>		Feathers the edges of the selection by the specified amount.

Method	Parameter type	Returns	What it does (Continued)
<b>fill</b> (filltype [, mode] [, opacity] [, preserveTransparency])	<a href="#">SolidColor</a> <a href="#">ColorBlendMode</a> number [1..100] boolean		Fills the selection.  opacity is a percentage value.
<b>grow</b> (tolerance, antiAlias)	number boolean		Grows the selection to include all adjacent pixels falling within the specified tolerance range.
<b>invert</b> ( )			Inverts the selection (deselects the selection and selects the rest of the layer or document).  <b>Tip:</b> To flip the selection shape, see <a href="#">rotate</a> .
<b>load</b> (from [, combination] [, inverting])	<a href="#">Channel</a> <a href="#">SelectionType</a> boolean		Loads the selection from the specified channel.
<b>makeWorkPath</b> ([tolerance])	number		Makes this selection item the work path for this document.
<b>resize</b> ([horizontal] [, vertical] [, anchor])	number number <a href="#">AnchorPosition</a>		Resizes the selected area to the specified dimensions and anchor position.
<b>resizeBoundary</b> ([horizontal] [, vertical] [, anchor])	number number <a href="#">AnchorPosition</a>		Changes the size of the selection to the specified dimensions around the specified anchor.
<b>rotate</b> (angle [, anchor])	number <a href="#">AnchorPosition</a>		Rotates the selection by the specified amount around the specified anchor point.
<b>rotateBoundary</b> (angle [, anchor])	number <a href="#">AnchorPosition</a>		Rotates the boundary of the selection around the specified anchor.
<b>select</b> (region [, type] [, feather] [, antiAlias])	array of number <a href="#">SelectionType</a> number boolean		Selects the specified region.  The region parameter is an array of four coordinates, [left, top, right, bottom].
<b>selectAll</b> ( )			Selects the entire layer.



Method	Parameter type	Returns	What it does (Continued)
<b>selectBorder</b> (width)	<a href="#">UnitValue</a>		Selects the selection border only (in the specified width); subsequent actions do not affect the selected area within the borders.
<b>similar</b> (tolerance, antiAlias)	number boolean		Grows the selection to include pixels throughout the image falling within the tolerance range.
<b>smooth</b> (radius)	number		Cleans up stray pixels left inside or outside a color-based selection (within the radius specified in pixels).
<b>store</b> (into [, combination])	<a href="#">Channel</a> <a href="#">SelectionType</a>		Saves the selection as a channel.
<b>stroke</b> (strokeColor, width [, location] [, mode] [, opacity] [, preserveTransparency])	<a href="#">SolidColor</a> number <a href="#">StrokeLocation</a> <a href="#">ColorBlendMode</a> number [1..100] boolean		Strokes the selection border. <code>opacity</code> is a percentage value.
<b>translate</b> ([deltaX] [, deltaY])	<a href="#">UnitValue</a> <a href="#">UnitValue</a>		Moves the entire selection relative to its current position.
<b>translateBoundary</b> ([deltaX] [, deltaY])	<a href="#">UnitValue</a> <a href="#">UnitValue</a>		Moves the selection relative to its current position.

## Selection sample script

The following script creates a checkerboard using the following steps:

- Create an 800 x 800 pixel document.
- Divide the entire document into 100 x 100 pixel squares.
- Select every other square in the first row, then shift the selection criteria to select the alternate squares in the following row. Repeat until every other square in the document is selected.
- Fill the selected squares with the foreground color from the palette.
- Invert the selection and fill the newly selected squares with the background color from the palette.
- Deselect the squares to remove the selection outlines (the "marching ants").

### Selection.jsx

```
// Save the current preferences
var startRulerUnits = app.preferences.rulerUnits
```

```
var startTypeUnits = app.preferences.typeUnits
var startDisplayDialogs = app.displayDialogs

// Set Adobe Photoshop CC 2015 to use pixels and display no dialogs
app.preferences.rulerUnits = Units.PIXELS
app.preferences.typeUnits = TypeUnits.PIXELS
app.displayDialogs = DialogModes.NO

//Close all the open documents
while (app.documents.length) {
    app.activeDocument.close()
}

//Create variables for the 800 pixel board divided in even 100 x 100 squares
var docSize = 800
var cells = 8
var cellSize = docSize / cells

// create a new document
var checkersDoc = app.documents.add(docSize, docSize, 72, "Checkers")

// Create a variable to use for selecting the checker board
// That allows me to shift the selection one square to the right
//on every other row, and then shift back for the rows in between.
var shiftIt = true

// loop through vertically to create the first row
for (var v = 0; v < docSize; v += cellSize) {

    // Switch the shift for a new row
    shiftIt = !shiftIt

    // loop through horizontally
    for (var h = 0; h < docSize; h += (cellSize * 2)) {

        // push over the cellSize to start with only
        if (shiftIt && h == 0) {
            h += cellSize
        }

        // Select a square
        selRegion = Array(Array(h, v),
            Array(h + cellSize, v),
            Array(h + cellSize, v + cellSize),
            Array(h, v + cellSize),
            Array(h, v))

        // In the first iteration of the loop, start the selection
        //In subsequent iterations, use the EXTEND constant value
        //of the select() method to add to the selection (in the loop's else clause)
        if (h == 0 && v == 0) {
            checkersDoc.selection.select(selRegion)
        } else {
            checkersDoc.selection.select(selRegion, SelectionType.EXTEND)
        }

        // turn this off for faster execution
        // turn this on for debugging
        WaitForRedraw()
    }
}
```

```
}

// Fill the current selection with the foreground color
checkersDoc.selection.fill(app.foregroundColor)

//Invert the selection
checkersDoc.selection.invert()

// Fill the new selection with the background color
checkersDoc.selection.fill(app.backgroundColor)

// Clear the selection to get rid of the non-printing borders
checkersDoc.selection.deselect()

// Reset the application preferences
app.preferences.rulerUnits = startRulerUnits
app.preferences.typeUnits = startTypeUnits
app.displayDialogs = startDisplayDialogs

// A helper function for debugging
// It also helps the user see what is going on
// if you turn it off for this example you
// get a flashing cursor for a number time
function WaitForRedraw()
{
    var eventWait = charIDToTypeID("Wait")
    var enumRedrawComplete = charIDToTypeID("RdCm")
    var typeState = charIDToTypeID("Stte")
    var keyState = charIDToTypeID("Stte")

    var desc = new ActionDescriptor()

    desc.putEnumerated(keyState, typeState, enumRedrawComplete)

    executeAction(eventWait, desc, DialogModes.NO)
}
```

## SGIRGBSaveOptions

Options for saving a document in SGIRGB format using the [Document.saveAs\(\)](#) method.

**Note:** The SGIRGB format is not installed automatically with Adobe Photoshop CC 2015.

### Properties

Property	Value type	What it is
<b>alphaChannels</b>	boolean	Read-write. True to save the alpha channels.
<b>spotColors</b>	boolean	Read-write. True to save the spot colors.
<b>typename</b>	string	Read-only. The class name of the referenced <code>SGIRGBSaveOptions</code> object.

# SolidColor

A color definition used in the document. Maps a color to equivalents in all available color models.

- Used in [Application.backgroundColor](#) and [foregroundColor](#) properties, in [Channel.color](#), in [ColorSampler.color](#), and in [TextItem.color](#)
- Passed to [PathItem.fillPath\(\)](#), [Selection.fill\(\)](#), and [Selection.stroke\(\)](#).

## Properties

Property	Value type	What it is
<b>cmyk</b>	<a href="#">CMYKColor</a>	Read-write. The CMYK color mode.
<b>gray</b>	<a href="#">GrayColor</a>	Read-write. The Grayscale color mode.
<b>hsb</b>	<a href="#">HSBColor</a>	Read-write. The HSB color mode.
<b>lab</b>	<a href="#">LabColor</a>	Read-write. The LAB color mode.
<b>model</b>	<a href="#">ColorModel</a>	Read-write. The color model.
<b>nearestWebColor</b>	<a href="#">RGBColor</a>	Read-only. The nearest web color to the current color.
<b>rgb</b>	<a href="#">RGBColor</a>	Read-write. The RGB color mode.
<b>typename</b>	string	Read-only. The class name of the referenced <code>SolidColor</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>isEqual</b> (color)	<a href="#">SolidColor</a>	boolean	True if the <code>SolidColor</code> object is visually equal to the specified color.

## SubPathInfo

An array of [PathPoint](#) objects that describes a straight or curved segment of a path, used to create a [SubPathItem](#).

Pass an array of these objects to the [PathItems.add\(\)](#) method. This method creates a [SubPathItem](#) object for each `SubPathInfo` object, and creates and returns a new [PathItem](#) object for the path represented by all of the subpaths.

- Use `SubPathInfo` to create subpaths; the properties are writeable.
- Use the [SubPathItem](#) object to retrieve information about existing subpaths. The properties are read-only.

### Properties

Property	Value type	What it is
<code>closed</code>	boolean	Read-write. True if the path describes an enclosed area.
<code>entireSubPath</code>	array of <a href="#">PathPoint</a>	Read-write.
<code>operation</code>	<a href="#">ShapeOperation</a>	Read-write. The subpath's operation on other subpaths. Specifies how to combine the shapes if the destination path already has a selection.
<code>typename</code>	string	Read-only. The class name of the referenced <code>SubPathInfo</code> object.

## SubPathItem

Represents a subpath; a collection of subpaths make up a [PathItem](#).

Create these objects by passing [SubPathInfo](#) objects to the [PathItems.add\(\)](#) method. This method creates a `SubPathItem` object for each [SubPathInfo](#) object, and creates and returns a new [PathItem](#) object for the path represented by all of the subpaths. Access these objects in the [PathItem.subPathItems](#) collection.

- Use the [SubPathItem](#) object to retrieve information about existing subpaths. The properties are read-only.
- Use [SubPathInfo](#) to create subpaths; the properties are writeable.

### Properties

Property	Value type	What it is
<b>closed</b>	boolean	Read-only. True if the path is closed.
<b>operation</b>	<a href="#">ShapeOperation</a>	Read-only. How this object behaves when it intersects another <a href="#">SubPathItem</a> object. Specifies how to combine the shapes if the destination path already has a selection.
<b>parent</b>	<a href="#">PathItem</a>	Read-only. The object's container.
<b>pathPoints</b>	<a href="#">PathPoints</a>	Read-only. The <code>PathPoints</code> collection.
<b>typename</b>	string	Read-only. The class name of the referenced <code>SubPathItem</code> object.

## SubPathItems

A collection of [SubPathItem](#) objects that make up a [PathItem](#). Access this object in the [PathItem.subPathItems](#) collection property.

- Use [SubPathInfo](#) to create subpaths; the properties are writeable.
- Use the [SubPathItem](#) object to retrieve information about existing subpaths. The properties are read-only.

### Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the collection.
<b>parent</b>	<a href="#">PathItem</a>	Read-only. The containing path item.
<b>typename</b>	string	Read-only. The class name of the referenced <code>SubPathItems</code> object.



# TargaSaveOptions

Options for saving a document in TGA (Targa) format using the [Document.saveAs\(\)](#) method.

## Properties

Property	Value type	What it is
<b>alphaChannels</b>	boolean	Read-write. True to save the alpha channels.
<b>resolution</b>	<a href="#">TargaBitsPerPixels</a>	Read-write. The number of bits per pixel (default: <code>TargaBitsPerPixels.TWENTYFOUR</code> ).
<b>rleCompression</b>	boolean	Read-write. True to use RLE compression (default: <code>true</code> ).
<b>typename</b>	string	Read-only. The class name of the referenced <code>TargaSaveOptions</code> object.

# TextFont

Describes a font that is available to the application. Access this object in the [Application.fonts](#) collection. For example:

```
var myFont = app.fonts.getByName("ArialMT");
```

## Properties

Property	Value type	What it is
<b>family</b>	string	Read-only. The font family.
<b>name</b>	string	Read-only. The name of the font.
<b>parent</b>	<a href="#">Application</a>	Read-only. The containing application.
<b>postScriptName</b>	string	Read-only. The PostScript name of the font.
<b>style</b>	string	Read-only. The font style.
<b>typename</b>	string	Read-only. The class name of the referenced <code>TextFont</code> object.

# TextFonts

The collection of fonts available on your computer. Fonts are represented by [TextFont](#) objects. Access this object in the [Application.fonts](#) collection property. For example, this displays the number of available fonts:

```
alert (app.fonts.length) ;
```

## Properties

Property	Value type	What it is
<b>length</b>	number	Read-only. The number of elements in the collection.
<b>parent</b>	<a href="#">Application</a>	Read-only. The containing application.
<b>typename</b>	string	Read-only. The class name of the referenced <code>TextFonts</code> object.

## Methods

Method	Parameter type	Returns	What it does
<b>getByName</b> (name)	string	<a href="#">TextFont</a>	Gets the first element in the <code>TextFonts</code> collection with the provided name.

## TextItem

The text in an [ArtLayer](#) object whose [kind](#) property is `LayerKind.TEXT`. Access this object in the [ArtLayer.textItem](#) property. For example:

```
myLayers[i].textItem.contents = "Layer in " + textArray[i] + " Set Inside "
```

Many of the properties use the [UnitValue](#) type, which combines a measurement value with a measurement unit. For information about this type, see the *JavaScript Tools Guide*.

## Properties

Property	Value type	What it is
<b>alternateLigatures</b>	boolean	Read-write. True to use alternate ligatures.  <b>Note:</b> Alternate ligatures are the same as Discretionary Ligatures. See Adobe Photoshop CC 2015 Help for more information.
<b>antiAliasMethod</b>	<a href="#">AntiAlias</a>	Read-write. The method of anti aliasing to use.
<b>autoKerning</b>	<a href="#">AutoKernType</a>	Read-write. The auto kerning option to use.
<b>autoLeadingAmount</b>	number [0.01..5000.00]	Read-write. The percentage to use for auto (default) leading (in points).  Valid only when <a href="#">useAutoLeading</a> = <code>true</code> .
<b>baselineShift</b>	<a href="#">UnitValue</a>	Read-write. The unit value to use in the baseline offset of text.
<b>capitalization</b>	<a href="#">TextCase</a>	Read-write. The text case.
<b>color</b>	<a href="#">SolidColor</a>	Read-write. The text color.
<b>contents</b>	string	Read-write. The actual text in the layer.
<b>desiredGlyphScaling</b>	number [50..200]	Read-write. The desired amount by which to scale the horizontal size of the text letters. A percentage value; at 100, the width of characters is not scaled.  Valid only when <a href="#">justification</a> = <code>Justification.CENTERJUSTIFIED</code> , <code>FULLYJUSTIFIED</code> , <code>LEFTJUSTIFIED</code> , or <code>Justification.RIGHTJUSTIFIED</code> .  When used, the <a href="#">minimumGlyphScaling</a> and <a href="#">maximumGlyphScaling</a> values are also required.

Property	Value type	What it is (Continued)
<b>desiredLetterScaling</b>	number [100..500]	<p>Read-write. The amount of space between letters (at 0, no space is added between letters).</p> <p>Equivalent to <b>Letter Spacing</b> in the Justification dialog (Select <b>Justification</b> on the Paragraphs palette menu).</p> <p>Valid only when <a href="#">justification</a> = <code>Justification.CENTERJUSTIFIED</code>, <code>FULLYJUSTIFIED</code>, <code>LEFTJUSTIFIED</code>, or <code>Justification.RIGHTJUSTIFIED</code>.</p> <p>When used, the <a href="#">minimumLetterScaling</a> and <a href="#">maximumLetterScaling</a> values are also required.</p>
<b>desiredWordScaling</b>	number [0..1000]	<p>Read-write. The amount (percentage) of space between words (at 100, no additional space is added between words).</p> <p>Equivalent to <b>Word Spacing</b> in the Justification dialog (Select <b>Justification</b> on the Paragraphs palette menu).</p> <p>Valid only when <a href="#">justification</a> = <code>Justification.CENTERJUSTIFIED</code>, <code>FULLYJUSTIFIED</code>, <code>LEFTJUSTIFIED</code>, or <code>Justification.RIGHTJUSTIFIED</code>.</p> <p>When used, the <a href="#">minimumWordScaling</a> and <a href="#">maximumWordScaling</a> values are also required.</p>
<b>direction</b>	<a href="#">Direction</a>	Read-write. The text orientation.
<b>fauxBold</b>	boolean	<p>Read-write. True to use faux bold (default: <code>false</code>).</p> <p>Setting this to true is equivalent to selecting text and clicking Faux Bold in the Character palette.</p>
<b>fauxItalic</b>	boolean	<p>Read-write. True to use faux italic (default: <code>false</code>).</p> <p>Setting this to true is equivalent to selecting text and clicking Faux Italic in the Character palette.</p>
<b>firstLineIndent</b>	<a href="#">UnitValue</a> [-1296..1296] points	Read-write. The amount (unit value) to indent the first line of paragraphs.
<b>font</b>	string	Read-write. The text face of the character. Use the PostScript Name of the font. See <a href="#">TextFont</a> and use the <code>postScriptName</code> property.
<b>hangingPunctuation</b>	boolean	Read-write. True to use Roman hanging punctuation.
<b>height</b>	<a href="#">UnitValue</a> X	<p>Read-write. The height of the bounding box (unit value) for paragraph text.</p> <p>Valid only when <a href="#">kind</a> = <code>TextType.PARAGRAPHTEXT</code>.</p>
<b>horizontalScale</b>	number [0..1000]	Read-write. Character scaling (horizontal) in proportion to <a href="#">verticalScale</a> (a percentage value).

Property	Value type	What it is (Continued)
<b>hyphenateAfterFirst</b>	number [1..15]	Read-write. The number of letters after which hyphenation in word wrap is allowed.
<b>hyphenateBeforeLast</b>	number [1..15]	Read-write. The number of letters before which hyphenation in word wrap is allowed.
<b>hyphenateCapitalWords</b>	boolean	Read-write. True to allow hyphenation in word wrap of capitalized words.
<b>hyphenateWordsLongerThan</b>	number [2..25]	Read-write. The minimum number of letters a word must have in order for hyphenation in word wrap to be allowed.
<b>hyphenation</b>	boolean	Read-write. True to use hyphenation in word wrap.
<b>hyphenationZone</b>	<a href="#">UnitValue</a> [0..720] pica	Read-write. The distance at the end of a line that will cause a word to break in unjustified type.
<b>hyphenLimit</b>	number	Read-write. The maximum number of consecutive lines that can end with a hyphenated word.
<b>justification</b>	<a href="#">Justification</a>	Read-write. The paragraph justification.
<b>kind</b>	<a href="#">TextType</a>	Read-write. The text-wrap type.
<b>language</b>	<a href="#">Language</a>	Read-write. The language to use.
<b>leading</b>	<a href="#">UnitValue</a>	Read-write. The leading amount.
<b>leftIndent</b>	<a href="#">UnitValue</a> [-1296..1296] points	Read-write. The amount of space to indent text from the left.
<b>ligatures</b>	boolean	Read-write. True to use ligatures.
<b>maximumGlyphScaling</b>	number [50..200]	<p>Read-write. The maximum amount to scale the horizontal size of the text letters (a percentage value; at 100, the width of characters is not scaled).</p> <p>Valid only when <a href="#">justification</a> = <code>Justification.CENTERJUSTIFIED</code>, <code>FULLYJUSTIFIED</code>, <code>LEFTJUSTIFIED</code>, or <code>Justification.RIGHTJUSTIFIED</code>.</p> <p>When used, the <a href="#">minimumGlyphScaling</a> and <a href="#">desiredGlyphScaling</a> values are also required.</p>

Property	Value type	What it is (Continued)
<b>maximumLetterScaling</b>	number [100..500]	<p>Read-write. The maximum amount of space to allow between letters (at 0, no space is added between letters).</p> <p>Equivalent to <b>Letter Spacing</b> in the Justification dialog (Select <b>Justification</b> on the Paragraphs palette menu).</p> <p>Valid only when <a href="#">justification</a> = <code>Justification.CENTERJUSTIFIED</code>, <code>FULLYJUSTIFIED</code>, <code>LEFTJUSTIFIED</code>, or <code>Justification.RIGHTJUSTIFIED</code>.</p> <p>When used, the <a href="#">minimumLetterScaling</a> and <a href="#">desiredLetterScaling</a> values are also required.</p>
<b>maximumWordScaling</b>	number [0..1000]	<p>Read-write. The maximum amount of space to allow between words (a percentage value; at 100, no additional space is added between words).</p> <p>Equivalent to <b>Word Spacing</b> in the Justification dialog (Select <b>Justification</b> on the Paragraphs palette menu).</p> <p>Valid only when <a href="#">justification</a> = <code>Justification.CENTERJUSTIFIED</code>, <code>FULLYJUSTIFIED</code>, <code>LEFTJUSTIFIED</code>, or <code>Justification.RIGHTJUSTIFIED</code>.</p> <p>When used, the <a href="#">minimumWordScaling</a> and <a href="#">desiredWordScaling</a> values are also required.</p>
<b>minimumGlyphScaling</b>	number [50..200]	<p>Read-write. The minimum amount to scale the horizontal size of the text letters (a percentage value; at 100, the width of characters is not scaled).</p> <p>Valid only when <a href="#">justification</a> = <code>Justification.CENTERJUSTIFIED</code>, <code>FULLYJUSTIFIED</code>, <code>LEFTJUSTIFIED</code>, or <code>Justification.RIGHTJUSTIFIED</code>.</p> <p>When used, the <a href="#">maximumGlyphScaling</a> and <a href="#">desiredGlyphScaling</a> values are also required.</p>

Property	Value type	What it is (Continued)
<b>minimumLetterScaling</b>	number [100..500]	<p>Read-write. The minimum amount of space to allow between letters (a percentage value; at 0, no space is removed between letters).</p> <p>Equivalent to <b>Letter Spacing</b> in the Justification dialog (Select <b>Justification</b> on the Paragraphs palette menu).</p> <p>Valid only when <a href="#">justification</a> = <code>Justification.CENTERJUSTIFIED</code>, <code>FULLYJUSTIFIED</code>, <code>LEFTJUSTIFIED</code>, or <code>Justification.RIGHTJUSTIFIED</code>.</p> <p>When used, the <a href="#">maximumLetterScaling</a> and <a href="#">desiredLetterScaling</a> values are also required.</p>
<b>minimumWordScaling</b>	number [0..1000]	<p>Read-write. The minimum amount of space to allow between words (a percentage value; at 100, no additional space is removed between words).</p> <p>Equivalent to <b>Word Spacing</b> in the Justification dialog (Select <b>Justification</b> on the Paragraphs palette menu).</p> <p>Valid only when <a href="#">justification</a> = <code>Justification.CENTERJUSTIFIED</code>, <code>FULLYJUSTIFIED</code>, <code>LEFTJUSTIFIED</code>, or <code>Justification.RIGHTJUSTIFIED</code>.</p> <p>When used, the <a href="#">maximumWordScaling</a> and <a href="#">desiredWordScaling</a> values are also required.</p>
<b>noBreak</b>	boolean	<p>Read-write. True to disallow line breaks in this text.</p> <p><b>Tip:</b> When true for many consecutive characters, can prevent word wrap and thus may prevent some text from appearing on the screen.</p>
<b>oldStyle</b>	boolean	Read-write. True to use old style type.
<b>parent</b>	<a href="#">ArtLayer</a>	Read-write. The containing layer.
<b>position</b>	array of <a href="#">UnitValue</a>	<p>Read-write. The position of origin for the text. The array members specify the X and Y coordinates.</p> <p>Equivalent to clicking the text tool at a point in the document to create the point of origin for text.</p>
<b>rightIndent</b>	<a href="#">UnitValue</a> [-1296..1296] points	Read-write. The amount of space to indent text from the right.
<b>size</b>	<a href="#">UnitValue</a>	Read-write. The font size in <code>UnitValue</code> . NOTE: Type was <code>points</code> for CS3 and older..
<b>spaceAfter</b>	<a href="#">UnitValue</a> [-1296..1296] points	Read-write. The amount of space to use after each paragraph.



Property	Value type	What it is (Continued)
<b>spaceBefore</b>	<a href="#">UnitValue</a> [-1296..1296] points	Read-write. The amount of space to use before each paragraph.
<b>strikeThru</b>	<a href="#">StrikeThruType</a>	Read-write. The text strike-through option to use.
<b>textComposer</b>	<a href="#">TextComposer</a>	Read-write. The composition method to use to evaluate line breaks and optimize the specified hyphenation and justification options.  Valid only when <a href="#">kind</a> = <code>TextType.PARAGRAPHTEXT</code> .
<b>tracking</b>	number [-1000..10000]	Read-write. The amount of uniform spacing between multiple characters.  Tracking units are 1/1000 of an em space. The width of an em space is relative to the current type size. In a 1-point font, 1 em equals 1 point; in a 10-point font, 1 em equals 10 points. So, for example, 100 units in a 10-point font are equivalent to 1 point.
<b>typename</b>	string	Read-only. The class name of the referenced <code>textItem</code> object.
<b>underline</b>	<a href="#">UnderlineType</a>	Read-write. The text underlining options.
<b>useAutoLeading</b>	boolean	Read-write. True to use a font's built-in leading information.
<b>verticalScale</b>	number [0-1000]	Read-write. Vertical character scaling in proportion to <a href="#">horizontalScale</a> (a percentage value).
<b>warpBend</b>	number [-100..100]	Read-write. The warp bend percentage.
<b>warpDirection</b>	<a href="#">Direction</a>	Read-write. The warp direction.
<b>warpHorizontalDistortion</b>	number [-100..100]	Read-write. The horizontal distortion of the warp (a percentage value).
<b>warpStyle</b>	<a href="#">WarpStyle</a>	Read-write. The style of warp to use.
<b>warpVerticalDistortion</b>	number [-100..100]	Read-write. The vertical distortion of the warp (a percentage value).
<b>width</b>	<a href="#">UnitValue</a>	Read-write. The width of the bounding box for paragraph text.  Valid only when <a href="#">kind</a> = <code>TextType.PARAGRAPHTEXT</code> .

## Methods

Method	Parameter type	Returns	What it does
<b>convertToShape</b> ( )			Converts the text item and its containing layer to a fill layer with the text changed to a clipping path.
<b>createPath</b> ( )			Creates a clipping path from the outlines of the actual text items (such as letters or words).

# TiffSaveOptions

Options for saving a document in TIFF format using the [Document.saveAs\(\)](#) method.

## Properties

Property	Value type	What it is
<b>alphaChannels</b>	boolean	Read-write. True to save the alpha channels.
<b>annotations</b>	boolean	Read-write. True to save the annotations.
<b>byteOrder</b>	<a href="#">ByteOrder</a>	Read-write. The order in which the document's multibyte values are read (default: <code>ByteOrder.MACOS</code> in Mac OS, <code>ByteOrder.IBM</code> in Windows).
<b>embedColorProfile</b>	boolean	Read-write. True to embed the color profile in the document.
<b>imageCompression</b>	<a href="#">TIFFEncoding</a>	Read-write. The compression type (default: <code>TIFFEncoding.NONE</code> ).
<b>interleaveChannels</b>	boolean	Read-write. True if the channels in the image will be interleaved.
<b>jpegQuality</b>	number [0..12]	Read-write. The quality of the produced image, which is inversely proportionate to the amount of JPEG compression.  Valid only when <a href="#">imageCompression</a> = <code>TIFFEncoding.JPEG</code> .
<b>layerCompression</b>	<a href="#">LayerCompression</a>	Read-write. The method of compression to use when saving layers (as opposed to saving composite data).  Valid only when <a href="#">layers</a> = <code>true</code> .
<b>layers</b>	boolean	Read-write. True to save the layers.
<b>saveImagePyramid</b>	boolean	Read-write. True to preserve multi-resolution information (default: <code>false</code> ).
<b>spotColors</b>	boolean	Read-write. True to save the spot colors.
<b>transparency</b>	boolean	Read-write. True to save the transparency as an additional alpha channel when the file is opened in another application.
<b>typename</b>	string	Read-only. The class name of the referenced <code>TiffSaveOptions</code> object.

## UnitValue

ExtendScript defines the JavaScript class `UnitValue` to represent measurement values together with their measurement units; see ['JavaScript support in Adobe Photoshop CC 2015' on page 32](#). For references details of these classes, see the *JavaScript Tools Guide*.

## xmpMetadata

Camera RAW image XMP metadata.

For camera RAW image files, the XMP metadata is stored in a *sidecar* file, which is a file in the same folder as the RAW file with the same base name and an XMP extension.

### Properties

Property	Value type	What it is
<b>parent</b>	<a href="#">Document</a>	Read-only. The containing document.
<b>rawData</b>	string	Read-write. A string containing the XMP metadata in XML (RDF) format. See the <i>XMP Specification</i> for details of this format.
<b>typename</b>	string	Read-only. The class name of the referenced <code>xmpMetadata</code> object.

## JavaScript Reference Guide

### JavaScript Resource

This section describes the JavaScript resource that enables your JavaScripts to behave like a plug-in. This includes:

- the ability to specify a menu the script appears in as a command,
- a terminology resource so the script can function with the Action Manager, which allows your script to record and be automated by scripting parameters,
- a category to enable ordering and grouping of commands within menus, and
- an enable string that indicates whether the command is enabled or disabled given a set of conditions.
- The strings must be valid XML syntax. The "&" character will not work for example. Use the `&amp;` for example to get logical '&' in the enableinfo block.

### JavaScript resource syntax

The JavaScript Resource has an HTML-style syntax, with each `<tag>` matched by a closing `</tag>`. This resource needs to appear within comments ( `/* ... */` ) and should be defined at the top of your script file (within the first 10,240 characters of the file.)

Tag	Description
<code>&lt;javascriptresource&gt;</code>	The resource definition tag.
<code>&lt;name&gt;</code>	The command name that appears in the Photoshop menu. If this tag is not provided in the resource, the name of the command in the menu defaults to the name of the script.
<code>&lt;menu&gt;</code>	The menu the command appears in. If this tag is not provided, the command appears in the <b>File &gt; Scripts</b> menu.  <b>Note:</b> Currently the only supported values for <code>&lt;menu&gt;</code> are <code>automate</code> , <code>filter</code> and <code>help</code> . <code>automate</code> puts the script in the <b>File &gt; Automate</b> menu for example.
<code>&lt;about&gt;</code>	A string that appears in an About box, which the user can select from the <b>Help &gt; About Plug-in</b> menu.

Tag	Description
<enableinfo>	<p>A boolean expression that indicates whether the command is enabled in the menu. See <a href="#">Enable-info grammar</a>.</p> <p><b>Note:</b> If you provide this tag, the menu item is enabled if and only if there is at least one document open, and the boolean expression evaluates to true. If you always want the menu item enabled, do not use this tag.</p>
<eventid>	A unique string that identifies the event. Using a UUID will ensure that your script won't share this identifier with another script.
<category>	The category the command appears within in the menu. Used to group and order commands in the menu. Commands are placed in the menu alphabetically based on the string in <category>. If two commands use the same category, they are grouped together.
<terminology>	The terminology dictionary for the script to function with the Action Manager. See the <a href="#">Terminology dictionary</a> .

## Basic JavaScript resource example

This example shows a very basic <javascriptresource>. With this resource, the script can be executed by selecting the command **Add a Document**, which appears in the **Automate** menu. This command is enabled in the menu, provided at least one document is already open. If the user requests information about the script from the **About Plug-in** menu, the string contained in the <about> tag is displayed in a dialog box.

```

/*
  <javascriptresource>
  <name>Add a Document</name>
  <type>automate</type>
  <about>A short string providing information about the script.</about>
  <enableinfo>true</enableinfo>
  </javascriptresource>
*/

app.documents.add();

```

## Enable-info grammar

The <enableinfo> tag provides a boolean expression that, when evaluated, indicates whether the command is enabled in the menu. You can use this expression to enable or disable the menu item based on various characteristics of the document. The Enable Info grammar is as follows:

```

<booleanExpression> :=    <conjunction> { "|" <conjunction> }
<conjunction> :=          <relation> {"&&" <relation> }
<relation> :=             <equality> {<relationOperator><equality>}
<equality> :=             <simpleExpression> {<equalityOperator><simpleExpression>}
<simpleExpression> :=      <term> {<addOperator><term>}
<term> :=                 <factor> {<mulOperator><factor>}

```

```

<factor> :=          <integer> | <intrinsic> | <ident> |
                    "(" <booleanExpression ">" | "(" simpleExpression ")" |
                    "+" <factor> | "-" <factor> | "!" <factor>

<integer> :=          digit {digit}

<intrinsic> :=        <limitFunction> | <dimFunction> | <inFunction>

<limitFunction> :=    ( "min" | "max" ) "(" <simpleExpression> ","
                    <simpleExpression> { "," <simpleExpression> } ")"

<dimFunction> :=      "dim" "(" <simpleExpression> "," <simpleExpression> ")"

<ident> :=            (alpha | "_") {alpha | digit | "_" }

<mulOperator> :=      "*" | "/"

<addOperator> :=      "+" | "-"

<equalityOperator> :=  "==" | "!="

<relationOperator> := "<" | "<=" | ">=" | ">"

<inFunction> :=       "in" "(" <simpleExpression> { "," <simpleExpression> } ")"

```

Operator precedence is shown in the following table. Operators are listed with the highest order of precedence at the top of the table

Operator	Description
	Or
&&	And
+ -	Addition or subtraction
* /	Multiply or divide
< <= >= >	Less than, less than or equal, greater than or equal, greater than
== !=	Equals, or does not equal.
(..) in() max() min() unary + - !	Functions Unary operators: increment, decrement, not

The grammar provides variables and constants that you can use in the <enableinfo> expression. The following table provides a list of the constants that are available.

Constant Name	Description
true	Boolean true
false	Boolean false
BitmapMode	Bitmap mode.
GrayScaleMode	Grayscale mode, 8 bit depth.
IndexedMode	Indexed color mode.
RGBMode	RGB color mode.



Constant Name	Description
CMYKMode	CMYK color mode.
HSLMode	HSL color mode.
HSBMode	HSB color mode
MultiChannelMode	Multichannel mode.
DuotoneMode	Duotone mode.
LabMode	Lab color mode.
Gray16Mode	Grayscale mode, 16 bits per channel
RGB48Mode	RGB color mode, 16 bits per channel.
Lab48Mode	LAB mode, 16 bits per channel.
CMYK64Mode	CMYK mode, 16 bits per channel.
DeepMultichannelMode	Deep multichannel mode.
Duotone16Mode	Duotone mode, 16 bit depth.
RGB96Mode	RGB color mode, 32 bits per channel.
Gray32Mode	Grayscale mode, 32 bit depth.

The following table show the set of variables you can use in the <enableinfo> expression. The value of these variables is set based on the properties of the active document.

Variable Name	Description
PSHOP_ImageMode	Image mode of the active document.
PSHOP_ImageDepth	Depth of the active document.
PSHOP_HasLayerMask	Boolean indicating presence of layer mask.
PSHOP_HasSelectionMask	Boolean indicating presence of selection mask.
PSHOP_HasTransparencyMask	Boolean indicating presence of transparency mask.
PSHOP_NumTargetChannels	Number of target channels.
PSHOP_NumTrueChannels	Numer of image channels.
PSHOP_IsAdjustorSheet	Boolean
PSHOP_IsTargetComposite	Boolean indicating whether channels are flattened.
PSHOP_IsTargetSection	Boolean.
PSHOP_IsTargetVisible	Boolean.
PSHOP_ImageWidth	Width of the image.
PSHOP_ImageHeight	Height of the image.
PSHOP_TargetProtectFlags.	

## Undefined values in enable-info evaluation

If any arithmetic or relation operation contains an operand whose value is undefined, or a variable that is undefined, the result of that evaluation is `false`.

Boolean values are treated as in C/C++, where non-zero values are `true`, and zero is `false`, with the exception that an undefined value is also `false`.

## Using the "in" function

The `in` function (see `<inFunction>`) returns `true` if the first parameter is equal to at least one of the subsequent parameters. A typical use might be to see if the image mode of the active document is one of a set of image modes. For example:

```
in(PSHOP_ImageMode, RGBMode, CMYKMode, LabMode)
```

## Action Manager automation

For your script to be able to record scripting parameters and be automated by them, it requires the addition of two basic mechanisms:

- A *terminology dictionary* that maps your script's user interface to human readable text, providing text and type information for each parameter the script uses.
- Code to read parameter information when it comes from the Action Manager, rather than from the user-interface, and code to write parameter information to the Action Manager. This code uses the Action Manager classes [ActionDescriptor](#), [ActionList](#), and [ActionReference](#).

See `Conditional Mode Change.jsx` for an example of a script that can record and be automated by scripting parameters. This file can be found in the Presets/Scripts folder.

## Terminology dictionary

The JavaScript resource provides a `<terminology>` tag that allows you to provide the terminology dictionary for your script. The first step in creating a terminology dictionary is to review your script's user interface, and create human-readable strings for each element in your user interface.

For example, in the Conditional Mode Change command, the user interface requests a source mode and a target mode. Both source mode and target mode have several options. All of these elements of the user interface need to have entries in the terminology dictionary.

The terminology dictionary is created in a PDF dictionary format, with the following entries, and must have the following format in the `<javascriptresource>`:

```
<terminology><![CDATA[<<<
  /Version integer
  /Events <<event dictionary>>
  /Classes <<class dictionary>>
  /Enumerations <<enumeration dictionary>>
>>> ]]></terminology>
```

**Note:** The information in the terminology tag needs to be wrapped in a `CDATA` block so the xml parser will ignore `"/"` and other tags that appear in the terminology.

The definitions for events, classes and enumerations dictionaries are provided below.

The `/Events` dictionary contains an entry for each event:

```

/eventName [                                // Name used in string-based API
  (String event name)                       // required
  /direct parameter type                   // optional; if omitted, no parameter
  <<                                       // optional parameter dictionary
    /parameterName [                      // Name used in string-based API
      (String name)                       // required
      /parameter type                     // required
    ]
    ...                                    // other parameters
  >>
]

```

The `/Classes` dictionary contains an entry for each class:

```

/className [                                // Name used in string-based API
  (ZString class name)                   // required
  <<                                       // property dictionary
    /propertyName [                     // Name used in string-based API
      (String name)                     // required
      /property type                     // required
    ]
    ...                                    // other properties
  >>
]

```

The `/Enumerations` dictionary contains an entry for each enumerated type:

```

/enumTypeName                               // Name used in string-based API
<<
  /enumValue (String name)               // required
  ...
>>

```

## Value type definitions

For `/parameter type` and `/property type` definitions, you can use the Class and Enumeration type declarations you make in your own terminology dictionary, you can use declarations provided by Photoshop or you can use basic value types.

### Basic Value Types

The basic value types are shown in the following table:

Name	Code	Description
<code>typeInteger</code>	<code>'long'</code>	<code>int32</code>
<code>typeFloat</code>	<code>'doub'</code>	IEEE 64 bit double
<code>typeBoolean</code>	<code>'bool'</code>	TRUE OR FALSE.
<code>typeText</code>	<code>'TEXT'</code>	Block of any number of readable characters.
<code>typeAlias</code>	<code>'alis'</code>	Macintosh file system path.

Name	Code	Description
typePaths	'Pth '	Windows file system path.
typePlatformFilePath	'alis' or 'Pth'	typeAlias for Mac OS, typePath for Windows.

### Predefined Class Types

Photoshop provides a number of predefined classes that are available for use in the terminology dictionary. A useful subset of those classes is shown in the table below. Use these classes when they are appropriate, but you can define new classes in the terminology resource, if necessary.

Name	Code	Description
classColor	'Clr '	Class for color classes.
classRGBColor	'RGBC'	keyRed, keyGreen, keyBlue
classCMYKColor	'CMYC'	keyCyan, keyMagenta, keyYellow, keyBlack.
classUnspecifiedColor	'UnsC'	Unspecified.
classGrayscale	'Grsc'	keyGray
classBookColor	'BkCl '	Book color
classLabColor	'LbCl '	keyLuminance, keyA, keyB.

## Uniqueness rules for terminology entries

Generally, the names for terminology entries must be unique within a particular category and scope. It is best to not make names unique unnecessarily; generic terms are preferable, and if a name already exists for something, go ahead and use it. Case matters in considering uniqueness of terminology entries.

The uniqueness rules for terminology entries are:

- All event names must be different from all other event names.
- All class names must be different from all other names.
- All enumeration type names must be different from all other enumeration type names.
- All keys must be different from all other keys used in the same class or event.
- All enumeration values must be different from all other enumeration values in the same enumeration type.
- A class, event, enumeration type, key, and enumeration value can all have the same name.

## Terminology definition example

This example demonstrates the terminology definition for a new event; the example uses ZStrings. The event is called `newAnnot`, and it takes three parameters:

- `annotType`, an enumeration (`annotType`)
- `at`, a class (`point`), and
- `size`, a class (`annotSizeClass`).

The `annotSizeClass` has two properties: `width`, and `height`, both of type `floatType`. The enumeration `annotType` has three values: `annotUnknown`, `annotText`, and `annotSound`.

```
<terminology><![CDATA[<<<
  /Version 1
  /Events
    /newAnnot [(New Annotation) <<
      /annotType [(Type) /annotType]
      /at [(At) /Point]
      /size [(Size) /annotSizeClass] >>]
  /Classes
    /annotSizeClass [(Size) <<
      /width [(Width) /floatType]
      /height [(Height) /floatType]
      >>]
  /Enumerations
    /annotType <<
      /annotUnknown (Unknown)
      /annotText (Text)
      /annotSound (Sound)
    >>> ]]></terminology>
```

# 4

## Scripting Constants

This section lists and describes the enumerations defined for use with Adobe Photoshop CC 2015 JavaScript properties and methods.

Constant type	Values	What it means
<b>AdjustmentReference</b>	ABSOLUTE RELATIVE	Method to use for interpreting selective color adjustment specifications:  ABSOLUTE = % of the whole.  RELATIVE = % of the existing color amount.  Pass to <a href="#">ArtLayer.selectiveColor()</a> .
<b>AnchorPosition</b>	BOTTOMCENTER BOTTOMLEFT BOTTOMRIGHT MIDDLECENTER MIDDLELEFT MIDDLERIGHT TOPCENTER TOPLEFT TOPRIGHT	The point around which to transform an object. This is the point that does not move when an object is rotated or resized using methods in <a href="#">ArtLayer</a> , <a href="#">LayerSet</a> , and <a href="#">Selection</a> , or when the entire canvas is resized with <a href="#">Document.resizeCanvas()</a> .
<b>AntiAlias</b>	CRISP NONE SHARP SMOOTH STRONG	Method to use to smooth edges by softening the color transition between edge pixels and background pixels. Used in a <a href="#">TextItem.antiAliasMethod</a> .
<b>AutoKernType</b>	MANUAL METRICS OPTICAL	The type of kerning to use for characters. Used in <a href="#">TextItem.autoKerning</a> .
<b>BatchDestinationType</b>	FOLDER NODESTINATION SAVEANDCLOSE	The destination, if any, for batch-processed files, specified in the <a href="#">BatchOptions</a> used with the <a href="#">Application.batch()</a> method:  FOLDER: Save modified versions of the files to a new location (leaving the originals unchanged).  NODESTINATIONTYPE: Leave all files open.  SAVEANDCLOSE: Save changes and close the files.

Constant type	Values	What it means
<b>BitmapConversionType</b>	CUSTOMPATTERN DIFFUSIONDITHER HALFTHRESHOLD HALFTONESCREEN PATTERNDITHER	Specifies the quality of an image you are converting to bitmap mode. Used in <a href="#">BitmapConversionOptions</a> .
<b>BitmapHalfToneType</b>	CROSS DIAMOND ELLIPSE LINE ROUND SQUARE	Specifies the shape of the dots (ink deposits) in the halftone screen. Used in <a href="#">BitmapConversionOptions</a> .
<b>BitsPerChannelType</b>	EIGHT ONE SIXTEEN THIRTYTWO	The number of bits per color channel. Value of <a href="#">Document.bitsPerChannel</a> ; pass to <a href="#">Documents.add()</a> . Also used in <a href="#">PDFOpenOptions</a> and <a href="#">CameraRAWOpenOptions</a> .
<b>BlendMode</b>	COLORBLEND COLORBURN COLORDODGE DARKEN DIFFERENCE DISSOLVE DIVIDE EXCLUSION HARDLIGHT HARDMIX HUE LIGHTEN LINEARBURN LINEARDODGE LINEARLIGHT LUMINOSITY MULTIPLY NORMAL OVERLAY PASSTHROUGH PINLIGHT SATURATION SCREEN SOFTLIGHT SUBTRACT VIVIDLIGHT	Controls how pixels in an image are blended when a filter is applied. The value of <a href="#">ArtLayer.blendMode</a> and <a href="#">LayerSet.blendMode</a> .
<b>BMPDepthType</b>	BMP_A1R5G5B5 BMP_A4R4G4B4 BMP_A8R8G8B8 BMP_R5G6B5 BMP_R8G8B8 BMP_X1R5G5B5 BMP_X4R4G4B4 BMP_X8R8G8B8 EIGHT FOUR ONE SIXTEEN THIRTYTWO TWENTYFOUR	The number of bits per channel (also called pixel depth or color depth). The number selected indicates the exponent of 2. For example, a pixel with a bit-depth of <code>EIGHT</code> has $2^8$ , or 256, possible color values.  Used in <a href="#">BMPSaveOptions</a> .

Constant type	Values	What it means
<b>ByteOrder</b>	IBM MACOS	The platform-specific order in which multibyte values are read.
<b>CameraRAWSettingsType</b>	CAMERA CUSTOM SELECTEDIMAGE	The default CameraRaw settings to use: the camera settings, custom settings, or the settings of the selected image. Set in <a href="#">CameraRAWOpenOptions</a> .
<b>CameraRAWSize</b>	EXTRALARGE LARGE MAXIMUM MEDIUM MINIMUM SMALL	The camera RAW size type options:  EXTRALARGE=5120 x 4096 LARGE=4096 x 2731 MAXIMUM=6144 X 4096 MEDIUM=3072 x 2048 MINIMUM=1536 x 1024 SMALL=2048 x 1365  Set in <a href="#">CameraRAWOpenOptions</a> .
<b>ChangeMode</b>	BITMAP CMYK GRAYSCALE INDEXEDCOLOR LAB MULTICHANNEL RGB	The new color profile or mode for a document, specified in <a href="#">Document.changeMode()</a> .  <b>Note:</b> Color images must be changed to GRAYSCALE mode before you can change them to BITMAP mode.
<b>ChannelType</b>	COMPONENT MASKEDAREA SELECTEDAREA SPOTCOLOR	The type of a color channel:  COMPONENT: related to document color mode.  MASKEDAREA: Alpha channel where color indicates masked area.  SELECTEDAREA: Alpha channel where color indicates selected are.  SPOTCOLOR: Alpha channel to store a spot color.



Constant type	Values	What it means
<b>ColorBlendMode</b>	BEHIND CLEAR COLOR COLORBURN COLORDODGE DARKEN DARKERCOLOR DIFFERENCE DISSOLVE EXCLUSION HARDLIGHT HARDMIXBLEND HUE LIGHTEN LIGHTERCOLOR LINEARBURN LINEARDODGE LINEARLIGHT LUMINOSITY MULTIPLY NORMAL OVERLAY PINLIGHT SATURATION SCREEN SOFTLIGHT VIVIDLIGHT	The way color should be blended in a fill or stroke operation. Pass to <a href="#">PathItem.fillPath()</a> , <a href="#">Selection.fill()</a> , <a href="#">Selection.stroke()</a>
<b>ColorModel</b>	CMYK GRAYSCALE HSB LAB NONE RGB	The color model to use for a <a href="#">SolidColor</a> .
<b>ColorPicker</b>	ADOBE APPLE PLUGIN WINDOWS	The preferred color-selection tool, set in <a href="#">Preferences</a> .
<b>ColorProfileType</b>	CUSTOM NONE WORKING	The type of color profile used to manage this document, set in <a href="#">Document.colorProfileType</a> .
<b>ColorReductionType</b>	ADAPTIVE BLACKWHITE CUSTOM GRAYSCALE MACINTOSH PERCEPTUAL RESTRICTIVE SELECTIVE WINDOWS	The color reduction algorithm option for <a href="#">ExportOptionsSaveForWeb</a> .
<b>ColorSpaceType</b>	ADOBERGB COLORMATCHRGB PROPHOTORGB SRGB	The type of color space to use in <a href="#">CameraRAWOpenOptions</a> .
<b>CopyrightedType</b>	COPYRIGHTEDWORK PUBLICDOMAIN UNMARKED	The copyright status of a document. Used in <a href="#">DocumentPrintSettings.copyrighted</a> .

Constant type	Values	What it means
<b>CreateFields</b>	DUPLICATION INTERPOLATION	The method to use for creating fields. Pass to <a href="#">ArtLayer.applyDelInterlace()</a> .
<b>CropToType</b>	ARTBOX BLEEDBOX BOUNDINGBOX CROPBOX MEDIABOX TRIMBOX	The style to use when cropping a page in a PDF document. Set in <a href="#">PDFOpenOptions.cropPage</a> .
<b>DCSType</b>	COLORCOMPOSITE GRAYSCALECOMPOSITE NOCOMPOSITE	The type of composite DCS file to create with <a href="#">DCS1_SaveOptions</a> or <a href="#">DCS2_SaveOptions</a> :  COLORCOMPOSITE: Creates a color composite file in addition to DCS files.  GRAYSCALECOMPOSITE: Creates a grayscale composite file in addition to DCS files.  NOCOMPOSITE: Does not create a composite file.
<b>DepthMapSource</b>	IMAGEHIGHLIGHT LAYERMASK NONE TRANSPARENCYCHANNEL	The source to use for the depth map. Pass to <a href="#">ArtLayer.applyLensBlur()</a> .
<b>DescValueType</b>	ALIATYPE BOOLEAN CLASSType DOUBLETType ENUMERATEDType INTEGERTType LARGEINTEGERTType LISTType OBJECTType RAWType REFERENCETType STRINGType UNITDOUBLE	The value type of an action key, returned by <a href="#">ActionDescriptor.getType()</a> and <a href="#">ActionList.getType()</a> .
<b>DialogModes</b>	ALL ERROR NO	Controls the type of dialogs Photoshop displays when running scripts.
<b>Direction</b>	HORIZONTAL VERTICAL	<ul style="list-style-type: none"> <li>• The direction in which to flip the document canvas, passed to <a href="#">Document.flipCanvas()</a>.</li> <li>• The orientation of text in <a href="#">TextItem.direction</a>.</li> <li>• The direction of text warping in <a href="#">TextItem.warpDirection</a>.</li> </ul>

Constant type	Values	What it means
<b>DisplacementMapType</b>	STRETCHTOFIT TILE	Describes how the displacement map fits the image if the image is not the same size as the map. Pass to <a href="#">ArtLayer.applyDisplace()</a> .
<b>Dither</b>	DIFFUSION NOISE NONE PATTERN	The type of dithering to use in <a href="#">GIFSaveOptions</a> , <a href="#">IndexedConversionOptions</a> and <a href="#">ExportOptionsSaveForWeb</a> .
<b>DocPositionStyle</b>	PRINTCENTERED USERDEFINED	The type of positioning to use in DocPosition
<b>DocumentFill</b>	BACKGROUNDCOLOR TRANSPARENT WHITE	The fill type of a new document, passed to <a href="#">Documents.add()</a> .
<b>DocumentMode</b>	BITMAP CMYK DUOTONE GRAYSCALE INDEXEDCOLOR LAB MULTICHANNEL RGB	The color mode of a open document, <a href="#">Document.mode</a> . See also <a href="#">Document.changeMode()</a> .
<b>EditLogItemsType</b>	CONCISE DETAILED SESSIONONLY	The preferred level of detail in th history log, set in <a href="#">Preferences</a> :  CONCISE: Save a concise history log.  DETAILED: Save a detailed history log.  SESSIONONLY: Save history log only for the session.
<b>ElementPlacement</b>	INSIDE PLACEATBEGINNING PLACEATEND PLACEBEFORE PLACEAFTER	The object's position in the Layers palette.  <b>Note:</b> Not all values are valid for all object types. See the specific object description to make sure you are using a valid value.
<b>EliminateFields</b>	EVENFIELDS ODDFIELDS	The type of fields to eliminate. Pass to <a href="#">ArtLayer.applyDelInterlace()</a> .
<b>ExportType</b>	ILLUSTRATORPATHS SAVEFORWEB	The type of export for <a href="#">Document.exportDocument()</a> .  This is equivalent to choosing <b>File &gt; Export &gt; Paths To Illustrator</b> , or <b>File &gt; Save For Web and Devices</b> .
<b>Extension</b>	LOWERCASE NONE UPPERCASE	The policy and format for appending an extension to the filename when saving with <a href="#">Document.saveAs()</a> .

Constant type	Values	What it means
<b>FileNamingType</b>	DDMM DDMMYY DOCUMENTNAMELOWER DOCUMENTNAMEMIXED DOCUMENTNAMEUPPER EXTENSIONLOWER EXTENSIONUPPER MMDD MMDDYY SERIALLETTERLOWER SERIALLETTERUPPER SERIALNUMBER1 SERIALNUMBER2 SERIALNUMBER3 SERIALNUMBER4 YYDDMM YYMMDD YYYYMMDD	File naming options for the <a href="#">BatchOptions</a> used with the <a href="#">Application.batch()</a> method.
<b>FontPreviewType</b>	HUGE EXTRALARGE LARGE MEDIUM NONE SMALL	The preferred type size to use for font previews in the type tool font menus , set in <a href="#">Preferences</a> .
<b>FontSize</b>	LARGE MEDIUM SMALL	The preferred type size to use for panels and dialogs, set in <a href="#">Preferences</a> .
<b>ForcedColors</b>	BLACKWHITE NONE PRIMARIES WEB	<p>The type of colors to be included the color table regardless of their usage. Used in <a href="#">GIFSaveOptions</a> and <a href="#">IndexedConversionOptions</a>.</p> <p>BLACKWHITE: Pure black and pure white.</p> <p>NONE: None</p> <p>PRIMARIES: Red, green, blue, cyan, magenta, yellow, black, and white.</p> <p>WEB: the 216 web-safe colors.</p>
<b>FormatOptions</b>	OPTIMIZEDBASELINE PROGRESSIVE STANDARDBASELINE	<p>The option with which to save a JPEG file, in <a href="#">JPEGSaveOptions</a>.</p> <p>OPTIMIZEDBASELINE: Optimized color and a slightly reduced file size.</p> <p>PROGRESSIVE: Displays a series of increasingly detailed scans as the image downloads.</p> <p>STANDARDBASELINE: Format recognized by most web browsers.</p>
<b>GalleryConstrainType</b>	CONSTRAINBOTH CONSTRAINHEIGHT CONSTRAINWIDTH	The type of proportions to constrain for images. Used in <a href="#">GalleryImagesOptions</a> .

Constant type	Values	What it means
<b>GalleryFontType</b>	ARIAL COURIERNEW HELVETICA TIMESNEWROMAN	The fonts to use for the Web photo gallery captions and other text. Used in <a href="#">GalleryBannerOptions</a> , <a href="#">GalleryImagesOptions</a> , and <a href="#">GalleryThumbnailOptions</a> .  Also used in <a href="#">PicturePackageOptions</a> .
<b>GallerySecurityTextColorType</b>	BLACK CUSTOM WHITE	The color to use for text displayed over gallery images as an antitheft deterrent. Used in <a href="#">GallerySecurityOptions</a> .
<b>GallerySecurityTextPositionType</b>	CENTERED LOWERLEFT LOWERRIGHT UPPERLEFT UPPERRIGHT	The position of the text displayed over gallery images as an antitheft deterrent. Used in <a href="#">GallerySecurityOptions</a> .  Also used in <a href="#">PicturePackageOptions</a> .
<b>GallerySecurityTextRotateType</b>	CLOCKWISE45 CLOCKWISE90 COUNTERCLOCKWISE45 COUNTERCLOCKWISE90 ZERO	The orientation of the text displayed over gallery images as an antitheft deterrent. Used in <a href="#">GallerySecurityOptions</a> .  Also used in <a href="#">PicturePackageOptions</a> .
<b>GallerySecurityType</b>	CAPTION COPYRIGHT CREDIT CUSTOMTEXT FILENAME NONE TITLE	The content to use for text displayed over gallery images as an antitheft deterrent. Used in <a href="#">GallerySecurityOptions</a> .  <b>Note:</b> All types draw from the image's file information except CUSTOMTEXT.
<b>GalleryThumbSizeType</b>	CUSTOM LARGE MEDIUM SMALL	The size of thumbnail images in the web photo gallery. Used in <a href="#">GalleryThumbnailOptions</a> .
<b>Geometry</b>	HEPTAGON HEXAGON OCTAGON PENTAGON SQUARE TRIANGLE	Geometric options for shapes, such as the iris shape in the Lens Blur Filter.  Pass to <a href="#">ArtLayer.applyLensBlur()</a> .
<b>GridLineStyle</b>	DASHED DOTTED SOLID	The preferred line style for the nonprinting grid displayed over images, set in <a href="#">Preferences</a> .
<b>GridSize</b>	LARGE MEDIUM NONE SMALL	The preferred size of grid line spacing, set in <a href="#">Preferences</a> .

Constant type	Values	What it means
<b>GuideLineStyle</b>	DASHED SOLID	The preferred line style for nonprinting guides displayed over images, set in <a href="#">Preferences</a> .
<b>IllustratorPathType</b>	ALLPATHS DOCUMENTBOUNDS NAMEDPATH	The paths to export to an Illustrator file using <a href="#">Document.exportDocument()</a> .
<b>Intent</b>	ABSOLUTECOLORIMETRIC PERCEPTUAL RELATIVECOLORIMETRIC SATURATION	The rendering intent to use when converting from one color space to another with <a href="#">Document.convertProfile()</a> or <a href="#">Document.print()</a>
<b>Justification</b>	CENTER CENTERJUSTIFIED FULLYJUSTIFIED LEFT LEFTJUSTIFIED RIGHT RIGHTJUSTIFIED	The placement of paragraph text within the bounding box. Used in <a href="#">TextItem.justification</a> .
<b>Language</b>	BRAZILLIANPORTUGUESE CANADIANFRENCH DANISH DUTCH ENGLISHUK ENGLISHUSA FINNISH FRENCH GERMAN ITALIAN NORWEGIAN NYNORSKNORWEGIAN OLDGERMAN PORTUGUESE SPANISH SWEDISH SWISSGERMAN	The language to use for text. Used in <a href="#">TextItem.language</a> .
<b>LayerCompression</b>	RLE ZIP	Compression methods for data for pixels in layers, when saving to TIFF format. Used in <a href="#">TiffSaveOptions</a> .

Constant type	Values	What it means
<b>LayerKind</b>	BLACKANDWHITE BRIGHTNESSCONTRAST CHANNELMIXER COLORBALANCE CURVES EXPOSURE GRADIENTFILL GRADIENTMAP HUESATURATION INVERSION LEVELS NORMAL PATTERNFILL PHOTOFILTER POSTERIZE SELECTIVECOLOR SMARTOBJECT SOLIDFILL TEXT THRESHOLD LAYER3D VIBRANCE VIDEO	The type of a layer object, in <a href="#">ArtLayer.kind</a> .  <b>Note:</b> You can create a text layer only from an empty art layer.
<b>LensType</b>	MOVIEPRIME PRIME105 PRIME35 ZOOMLENS	The type of lens to use. Pass to <a href="#">ArtLayer.applyLensFlare()</a> .
<b>MagnificationType</b>	ACTUALSIZE FITPAGE	The type of magnification to use when viewing an image. Used in <a href="#">PresentationOptions</a> .
<b>MatteType</b>	BACKGROUND BLACK FOREGROUND NETSCAPE NONE SEMIGRAY WHITE	The color to use to fill anti-aliased edges adjacent to transparent areas of the image. When transparency is turned off for an image, the matte color is applied to transparent areas.  Used in <a href="#">GIFSaveOptions</a> , <a href="#">IndexedConversionOptions</a> , and <a href="#">JPEGSaveOptions</a> .
<b>MeasurementRange</b>	ALLMEASUREMENTS ACTIVEMEASUREMENTS	The measurement to act upon. Pass to <a href="#">MeasurementLog</a> methods.
<b>MeasurementSource</b>	MEASURESELECTION MEASURECOUNTOOL MEASURERULERTOOL	The source for recording measurements. Pass to <a href="#">Document.recordMeasurements()</a> .
<b>NewDocumentMode</b>	BITMAP CMYK GRAYSCALE LAB RGB	The color profile to use for a new document.  Pass to <a href="#">Documents.add()</a> .  Also used in <a href="#">ContactSheetOptions</a> and <a href="#">PicturePackageOptions</a> .

Constant type	Values	What it means
<b>NoiseDistribution</b>	GAUSSIAN UNIFORM	Distribution method to use when applying an Add Noise filter. Pass to <a href="#">ArtLayer.applyAddNoise()</a> .
<b>OffsetUndefinedAreas</b>	REPEATEDGEPIXELS SETTOBACKGROUND WRAPAROUND	Method to use to fill the empty space left by offsetting a an image or selection. Pass to <a href="#">ArtLayer.applyOffset()</a> .
<b>OpenDocumentMode</b>	CMYK GRAYSCALE LAB RGB	The color profile to use when opening an EPS or PDF document. Pass to <code>app.open()</code> in <a href="#">EPSOpenOptions</a> or <a href="#">PDFOpenOptions</a> .
<b>OpenDocumentType</b>	ALIASPIX BMP CAMERARAW COMPUSERVEGIF DICOM ELECTRICIMAGE EPS EPSPICTPREVIEW EPSTIFFPREVIEW FILMSTRIP JPEG PCX PDF PHOTOCOD PHOTOSHOP PHOTOSHOPDCS_1 PHOTOSHOPDCS_2 PHOTOSHOPEPS PHOTOSHOPPDF PICTFILEFORMAT PICTRESOURCEFORMAT PIXAR PNG PORTABLEBITMAP RAW SCITEXCT SGIRGB SOFTIMAGE TARGA TIFF WAVEFRONTTLA WIRELESSBITMAP	<p>The format in which to open the document, using <code>app.open()</code>.</p> <p><b>Note:</b> PHOTOCOD is deprecated. Kodak PhotoCD is now found in the Goodies folder on the Adobe Photoshop CC 2015 Install DVD.</p> <p><b>Note:</b> The DICOM option is for the Extended version only.</p>
<b>OperatingSystem</b>	OS2 WINDOWS	The target operating system in <a href="#">BMPSaveOptions</a> .
<b>Orientation</b>	LANDSCAPE PORTRAIT	<p>Page orientation for <a href="#">PhotoCDOpenOptions</a>, deprecated in Photoshop CS3.</p> <p><b>Note:</b> Kodak PhotoCD is now found in the Goodies folder on the Adobe Photoshop CC 2015 Install DVD.</p>



Constant type	Values	What it means
<b>OtherPaintingCursors</b>	PRECISEOTHER STANDARDOTHER	The preferred pointer for the following tools: Eraser, Pencil, Paintbrush, Healing Brush, Rubber Stamp, Pattern Stamp, Smudge, Blur, Sharpen, Dodge, Burn, Sponge. Set in <a href="#">Preferences</a> .
<b>PaintingCursors</b>	BRUSHSIZE PRECISE STANDARD	The preferred pointer for the following tools: Marquee, Lasso, Polygonal Lasso, Magic Wand, Crop, Slice, Patch Eyedropper, Pen, Gradient, Line, Paint Bucket, Magnetic Lasso, Magnetic Pen, Freeform Pen, Measure, Color Sampler. Set in <a href="#">Preferences</a> .
<b>PaletteType</b>	EXACT LOCALADAPTIVE LOCALPERCEPTUAL LOCALSELECTIVE MACOSPALETTE MASTERADAPTIVE MASTERPERCEPTUAL MASTERSELECTIVE PREVIOUSPALETTE UNIFORM WEBPALETTE WINDOWSPALETTE	The palette type to use in <a href="#">GIFSaveOptions</a> and <a href="#">IndexedConversionOptions</a> .
<b>PathKind</b>	CLIPPINGPATH NORMALPATH TEXTMASK VECTORMASK WORKPATH	The type of a <a href="#">PathItem</a> .
<b>PDFCompatibility</b>	PDF13 PDF14 PDF15 PDF16 PDF17	The PDF version to make the document compatible with. Used in <a href="#">PDFSaveOptions</a> .
<b>PDFEncoding</b>	JPEG JPEG2000HIGH JPEG2000LOSSLESS JPEG2000LOW JPEG2000MED JPEG2000MEDHIGH JPEG2000MEDLOW JPEGHIGH JPEGLow JPEGMED JPEGMEDHIGH JPEGMEDLOW NONE PDFZIP PDFZIP4BIT	The type of compression to use when saving a document in PDF format. Used in <a href="#">PDFSaveOptions</a> .

Constant type	Values	What it means
<b>PDFResample</b>	NONE PDFAVERAGE PDFBICUBIC PDFSUBSAMPLE	The down sample method to use. Used in <a href="#">PDFSaveOptions</a> .
<b>PDFStandard</b>	NONE PDFX1A2001 PDFX1A2003 PDFX32002 PDFX32003 PDFX42008	The PDF standard to make the document compatible with. Used in <a href="#">PDFSaveOptions</a> .
<b>PhotoCDColorSpace</b>	LAB16 LAB8 RGB16 RGB8	The color space for <a href="#">PhotoCDOpenOptions</a> , deprecated in Photoshop CS3.  <b>Note:</b> Kodak PhotoCD is now found in the Goodies folder on the Adobe Photoshop CC 2015 Install DVD.
<b>PhotoCDSize</b>	EXTRALARGE LARGE MAXIMUM MEDIUM MINIMUM SMALL	The pixel dimensions of the image in <a href="#">PhotoCDOpenOptions</a> , deprecated in Photoshop CS3.  EXTRALARGE = 1024x1536 LARGE = 512x768 MAXIMUM = 2048x3072 MEDIUM = 256x384 MINIMUM = 64x96 SMALL = 128x192  <b>Note:</b> Kodak PhotoCD is now found in the Goodies folder on the Adobe Photoshop CC 2015 Install DVD.
<b>PICTBitsPerPixels</b>	EIGHT FOUR SIXTEEN THIRTYTWO TWO	The number of bits per pixel to use when compression a PICT file. Used in <a href="#">PICTFileSaveOptions</a> and <a href="#">PICTResourceSaveOptions</a> .  <b>Note:</b> Use 16 or 32 for RGB images; use 2, 4, or 8 for bitmap and grayscale images.
<b>PICTCompression</b>	JPEGHIGHPICT JPEGLOWPICT JPEGMAXIMUMPICT JPEGMEDIUMPICT NONE	The type of compression to use when saving an image as a PICT file. Used in <a href="#">PICTFileSaveOptions</a> and <a href="#">PICTResourceSaveOptions</a> .
<b>PicturePackageTextType</b>	CAPTION COPYRIGHT CREDIT FILENAME NONE ORIGIN USER	The function or meaning of text in a Picture Package. Used in <a href="#">PicturePackageOptions</a> .

Constant type	Values	What it means
<b>PointKind</b>	CORNERPOINT SMOOTHPOINT	The role a <a href="#">PathPoint</a> plays in a <a href="#">PathItem</a> .
<b>PointType</b>	POSTSCRIPT TRADITIONAL	The preferred measurement to use for type points, set in <a href="#">Preferences.pointSize</a> :  POSTSCRIPT = 72 points/inch.  TRADITIONAL = 72.27 points/inch.
<b>PolarConversionType</b>	POLARTORECTANGULAR RECTANGULARTOPOLAR	The method of polar distortion to use. Pass to <a href="#">ArtLayer.applyPolarCoordinates()</a> .
<b>Preview</b>	EIGHTBITTIFF MACOSEIGHTBIT MACOSJPEG MACOSMONOCHROME MONOCHROMETIFF NONE	The type of image to use as a low-resolution preview in the destination application.  Used in <a href="#">DCS1_SaveOptions</a> , <a href="#">DCS2_SaveOptions</a> , and <a href="#">EPSSaveOptions</a> .
<b>PrintColorHandling</b>	PRINTERMANAGED PHOTOSHOPMANAGED SEPARATIONS	The type of color handling to use for ColorHandling
<b>PurgeTarget</b>	ALLCACHES CLIPBOARDCACHE HISTORYCACHES UNDOCACHES	Cache to be targeted in an <a href="#">Application.purge()</a> operation.
<b>QueryStateType</b>	ALWAYS ASK NEVER	The preferred policy for checking whether to maximize compatibility when opening PSD files, set in <a href="#">Preferences.maximizeCompatibility</a> .
<b>RadialBlurMethod</b>	SPIN ZOOM	The blur method to use. Pass to <a href="#">ArtLayer.applyRadialBlur()</a> .
<b>RadialBlurQuality</b>	BEST DRAFT GOOD	The smoothness or graininess of the blurred image. Pass to <a href="#">ArtLayer.applyRadialBlur()</a> .
<b>RasterizeType</b>	ENTIRELAYER FILLCONTENT LAYERCLIPPINGPATH LINKEDLAYERS SHAPE TEXTCONTENTS	The layer element to rasterize, using <a href="#">ArtLayer.rasterize()</a> .
<b>ReferenceFormType</b>	CLASSTYPE ENUMERATED IDENTIFIER INDEX NAME OFFSET PROPERTY	The type of an <a href="#">ActionReference</a> object, returned by <a href="#">getForm()</a> .

Constant type	Values	What it means
<b>ResampleMethod</b>	AUTOMATIC BICUBIC BICUBICAUTOMATIC BICUBICSHARPER BICUBICSMOOTHER BILINEAR NEARESTNEIGHBOR NONE PRESERVEDETAILS	The method to use for image interpolation. Passed to <a href="#">Document.resizeImage()</a> , and used as the value of <a href="#">Preferences.interpolation</a> .
<b>RippleSize</b>	LARGE MEDIUM SMALL	The size of undulations to use. Pass to <a href="#">ArtLayer.applyRipple()</a> .
<b>SaveBehavior</b>	ALWAYS SAVE ASKWHENSAVING NEVERSAVE	The application's preferred behavior when saving a document. See <a href="#">Preferences.appendExtension</a> and <a href="#">imagePreviews</a>
<b>SaveDocumentType</b>	ALIASPIX BMP COMPUSEVEGIF ELECTRICIMAGE JPEG PCX PHOTOSHOP PHOTOSHOPDCS_1 PHOTOSHOPDCS_2 PHOTOSHOPEPS PHOTOSHOPPDF PICTFileFORMAT PICTRESOURCEFORMAT PIXAR PNG PORTABLEBITMAP RAW SCITEXCT SGIRGB SOFTIMAGE TARGA TIFF WAVEFRONTFLA WIRELESSBITMAP	The format in which to save a document when exporting with <a href="#">Document.exportDocument()</a> . Pass in <a href="#">ExportOptionsSaveForWeb.format</a> , to specify the type of file to write. Only the following are supported for export: COMPUSEVEGIF, JPEG, PNG-8, PNG-24, and BMP.
<b>SaveEncoding</b>	ASCII BINARY JPEGHIGH JPEGLOW JPEGMAXIMUM JPEGMEDIUM	The type of encoding to use when saving a file to DCS or EPS with <a href="#">Document.saveAs()</a> .
<b>SaveLogItemsType</b>	LOGFILE LOGFILEANDMETADATA METADATA	The preferred location of history log data, set in <a href="#">Preferences.saveLogItems</a> .
<b>SaveOptions</b>	DONOTSAVECHANGES PROMPTTOSAVECHANGES SAVECHANGES	The policy for closing a document with <a href="#">Document.close()</a> .

Constant type	Values	What it means
<b>SelectionType</b>	DIMINISH EXTEND INTERSECT REPLACE	<p>The selection behavior when a selection already exists:</p> <p>DIMINISH: Remove the selection from the already selected area.</p> <p>EXTEND: Add the selection to an already selected area.</p> <p>INTERSECT: Make the selection only the area where the new selection intersects the already selected area.</p> <p>REPLACE: Replace the selected area.</p> <p>Used in <a href="#">PathItem.makeSelection()</a>, <a href="#">Selection.load()</a>, <a href="#">Selection.select()</a>, and <a href="#">Selection.store()</a>.</p>
<b>ShapeOperation</b>	SHAPEADD SHAPEINTERSECT SHAPESUBTRACT SHAPEXOR	<p>How to combine the shapes if the destination path already has a selection.</p> <p>Set for <a href="#">SubPathInfo.operation</a>, stored in the resulting <a href="#">SubPathItem</a>.</p>
<b>SmartBlurMode</b>	EDGEONLY NORMAL OVERLAYEDGE	<p>The method to use for smart blurring:</p> <p>EDGEONLY, OVERLAYEDGES: Apply blur only to edges of color transitions.</p> <p>NORMAL: Apply blur to entire image.</p> <p>Pass to <a href="#">ArtLayer.applySmartBlur()</a>.</p>
<b>SmartBlurQuality</b>	HIGH LOW MEDIUM	The blur quality to use. Pass to <a href="#">ArtLayer.applySmartBlur()</a> .
<b>SourceSpaceType</b>	DOCUMENT PROOF	The color space for source when printing with <a href="#">Document.print()</a> .
<b>SpherizeMode</b>	HORIZONTAL NORMAL VERTICAL	The curve (or stretch shape) to use for the distortion. Pass to <a href="#">ArtLayer.applySpherize()</a> .
<b>StrikeThruType</b>	STRIKEBOX STRIKEHEIGHT STRIKEOFF	The style of strikethrough to use in text. Used in <a href="#">TextItem.strikeThru</a> .
<b>StrokeLocation</b>	CENTER INSIDE OUTSIDE	The placement of path or selection boundary strokes. Pass to <a href="#">Selection.stroke()</a> .
<b>TargaBitsPerPixels</b>	SIXTEEN THIRTYTWO TWENTYFOUR	The resolution to use when saving an image in Targa format. Used in <a href="#">TargaSaveOptions</a> .

Constant type	Values	What it means
<b>TextCase</b>	ALLCAPS NORMAL SMALLCAPS	The capitalization style to use in text. Used in <a href="#">TextItem.capitalization</a> .
<b>TextComposer</b>	ADOBEEVERYLINE ADOBESINGLELINE	The composition method to use to optimize the specified hyphenation and justification options. Used in <a href="#">TextItem.textComposer</a> .
<b>TextType</b>	PARAGRAPHTEXT POINTTEXT	The type of text, used in <a href="#">TextItem.kind</a> .  PARAGRAPHTEXT: Text that wraps within a bounding box.  POINTTEXT: Text that does not wrap.
<b>TextureType</b>	BLOCKS CANVAS FILE FROSTED TINYLENS	The type of texture or glass surface image to load for a texturizer or glass filter. Pass to <a href="#">ArtLayer.applyGlassEffect()</a> .
<b>TIFFEncoding</b>	JPEG NONE TIFFLZW TIFFZIP	The type of compression to use for TIFF files. Used in <a href="#">TiffSaveOptions</a> .
<b>ToolType</b>	ARTHISTORYBRUSH BACKGROUNDeraser BLUR BRUSH BURN CLONESTAMP COLORREPLACEMENTTOOL DODGE ERASER HEALINGBRUSH HISTORYBRUSH PATTERNSTAMP PENCIL SHARPEN SMUDGE SPONGE	The tool to use with <a href="#">PathItem.strokePath()</a> .
<b>TransitionType</b>	BLINDSHORIZONTAL BLINDSVERTICAL BOXIN BOXOUT DISSOLVE GLITTERDOWN GLITTERRIGHT GLITTERRIGHTDOWN NONE RANDOM SPLITHORIZONTALIN SPLITHORIZONTALOUT SPLITVERTICALIN SPLITVERTICALOUT WIPEDOWN WIPELEFT WIPERIGHT WIPEUP	The method to use for transition from one image to the next in a PDF presentation. Used in <a href="#">PresentationOptions</a> .

Constant type	Values	What it means
<b>TrimType</b>	BOTTOMRIGHT TOPLEFT TRANSPARENT	Type of pixels to trim around an image, passed to <a href="#">Document.trim()</a> .:  BOTTOMRIGHT = bottom right pixel color.  TOPLEFT = top left pixel color.
<b>TypeUnits</b>	MM PIXELS POINTS	The preferred unit for text character measurements, set in <a href="#">Preferences</a> .
<b>UndefinedAreas</b>	REPEATEDGEPIXELS WRAPAROUND	The method to use to treat undistorted areas or areas left blank in an image to which the a filter in the Distort category has been applied. Pass to <a href="#">ArtLayer.applyDisplace()</a> , <a href="#">applyShear()</a> , <a href="#">applyWave()</a> .
<b>UnderlineType</b>	UNDERLINELEFT UNDERLINEOFF UNDERLINERIGHT	The placement of text underlining. Used in <a href="#">TextItem.underline</a> .  <b>Note:</b> UNDERLINELEFT and UNDERLINERIGHT are valid only when <code>direction = Direction.VERTICAL</code> .
<b>Units</b>	CM INCHES MM PERCENT PICAS PIXELS POINTS	The preferred measurement unit for type and ruler increments, set in <a href="#">Preferences.rulerUnits</a> .
<b>Urgency</b>	FOUR HIGH LOW NONE NORMAL SEVEN SIX THREE TWO	The editorial urgency status of a document, set in <a href="#">DocumentPrintSettings.urgency</a> .
<b>WarpStyle</b>	ARC ARCH ARCLOWER ARCUPPER BULGE FISH FISHEYE FLAG INFLATE NONE RISE SHELLLOWER SHELLUPPER SQUEEZE TWIST WAVE	The warp style to use for text. Used in <a href="#">TextItem.warpStyle</a> .

Constant type	Values	What it means
<b>WaveType</b>	SINE SQUARE TRIANGULAR	The type of wave to use. Pass to <a href="#">ArtLayer.applyWave()</a> .
<b>WhiteBalanceType</b>	ASSHOT AUTO CLOUDY CUSTOM DAYLIGHT FLASH FLUORESCENT SHADE TUNGSTEN	Lighting conditions that affect color balance. Set in <a href="#">CameraRAWOpenOptions</a> .
<b>ZigZagType</b>	AROUNDCENTER OUTFROMCENTER PONDRIPPLES	The method of zigzagging to use. Pass to <a href="#">ArtLayer.applyZigZag()</a> .



## Appendix A: Event ID Codes

The following table lists events and their four-character ID codes or string identifiers for use with the `notifier` object.

**Note:** Do not include single quotes ( ' ) with four-character IDs in your code. The single quotes are used in this table to illustrate the placement of required spaces in codes that do not contain four letters. However, string identifiers, which are longer than four characters, require double quotes in the code.

**Tip:** If you can't find the event you want to use for notification in this table, you can use `ScriptListener` to determine the event ID code. See the `ScriptListener` documentation in the Action Manager chapter of the *Photoshop CC 2015 Scripting Guide*.

Event	4-char ID or String
3DTransform	'TdT '
Average	'Avrg'
ApplyStyle	'ASty'
Assert	'Asrt'
AccentedEdges	'AccE'
Add	'Add '
AddNoise	'AdNs'
AddTo	'AddT'
Align	'Algn'
All	'All '
AngledStrokes	'AngS'
ApplyImage	'AppI'
BasRelief	'BsRl'
Batch	'Btch'
BatchFromDroplet	'BtcF'
Blur	'Blr '
BlurMore	'BlrM'
Border	'Brdr'
Brightness	'BrgC'
CanvasSize	'CnvS'
ChalkCharcoal	'ChlC'
ChannelMixer	'ChnM'

Event	4-char ID or String
Charcoal	'Chrc'
Chrome	'Chrm'
Clear	'Cler'
Close	'Cls '
Clouds	'Clds'
ColorBalance	'ClrB'
ColorHalftone	'ClrH'
ColorRange	'ClrR'
ColoredPencil	'ClrP'
ContactSheet	"0B71D221-F8CE-11d2-B21B-0008C75B322C"
ConteCrayon	'CntC'
Contract	'Cntc'
ConvertMode	'CnvM'
Copy	'copy'
CopyEffects	'CpFX'
CopyMerged	'CpyM'
CopyToLayer	'CpTL'
Craquelure	'Crql'
CreateDroplet	'CrtD'
Crop	'Crop'
Crosshatch	'Crsh'
Crystallize	'Crst'
Curves	'Crvs'
Custom	'Cstm'
Cut	'cut '
CutToLayer	'CtTL'
Cutout	'Ct '
DarkStrokes	'DrkS'
DeInterlace	'Dntr'
DefinePattern	'DfnP'
Defringe	'Dfrg'
Delete	'Dlt '
Desaturate	'Dstt'

Event	4-char ID or String
Deselect	'Dslc'
Despeckle	'Dspc'
DifferenceClouds	'DrfC'
Diffuse	'Dfs '
DiffuseGlow	'DfsG'
DisableLayerFX	'dlfx'
Displace	'Dspl'
Distribute	'Dstr'
Draw	'Draw'
DryBrush	'DryB'
Duplicate	'Dplc'
DustAndScratches	'DstS'
Emboss	'Embs'
Equalize	'Eqlz'
Exchange	'Exch'
Expand	'Expn'
Export	'Expr'
JumpTo	'Jpto'
ExportTransparentImage	"02879e00-cb66-11d1-bc43-0060b0a13dc4"
Extrude	'Extr'
Facet	'Fct '
Fade	'Fade'
Feather	'Fthr'
Fibers	'Fbrs'
Fill	'Fl '
FilmGrain	'FlmG'
Filter	'Fltr'
FindEdges	'FndE'
FitImage	"3caa3434-cb67-11d1-bc43-0060b0a13dc4"
FlattenImage	'FltI'
Flip	'Flip'
Fragment	'Frgm'
Fresco	'Frsc'

Event	4-char ID or String
GaussianBlur	'GsnB'
Get	'getd'
Glass	'Gls '
GlowingEdges	'GlwE'
Gradient	'Grdn'
GradientMap	'GrMp'
Grain	'Grn '
GraphicPen	'GraP'
Group	'GrpL'
Grow	'Grow'
HalftoneScreen	'Hlfs'
Hide	'Hd '
HighPass	'HghP'
HSBHSL	'HsbP'
HueSaturation	'HStr'
ImageSize	'ImgS'
Import	'Impr'
InkOutlines	'InkO'
Intersect	'Intr'
IntersectWith	'IntW'
Inverse	'Invs'
Invert	'Invr'
LensFlare	'LnsF'
Levels	'Lvls'
LightingEffects	'LghE'
Link	'Lnk '
Make	'Mk '
Maximum	'Mxm '
Median	'Mdn '
MergeLayers	'Mrg2'
MergeLayersOld	'MrgL'
MergeSpotChannel	'MSpt'
MergeVisible	'MrgV'

Event	4-char ID or String
Mezzotint	'Mztn'
Minimum	'Mnm '
ModeChange	"8cba8cd6-cb66-11d1-bc43-0060b0a13dc4"
Mosaic	'Msc '
Mosaic_PLUGIN	'MscT'
MotionBlur	'MtnB'
Move	'move'
NTSCColors	'NTSC'
NeonGlow	'NGLw'
Next	'Nxt '
NotePaper	'NtPr'
Notify	'Ntfy'
Null	typeNull
OceanRipple	'OcnR'
Offset	'Ofst'
Open	'Opn '
Paint	'Pnt '
PaintDaubs	'PntD'
PaletteKnife	'PltK'
Paste	'past'
PasteEffects	'PaFX'
PasteInto	'PstI'
PasteOutside	'PstO'
Patchwork	'Ptch'
Photocopy	'Phtc'
PicturePackage	"4C1ABF40-DD82-11d2-B20F-0008C75B322C"
Pinch	'Pnch'
Place	'Plc '
Plaster	'Plst'
PlasticWrap	'PlsW'
Play	'Ply '
Pointillize	'Pntl'
Polar	'Plr '

Event	4-char ID or String
PosterEdges	'PstE'
Posterize	'Pstr'
Previous	'Prvs'
Print	'Prnt'
ProfileToProfile	'PrfT'
Purge	'Prge'
Quit	'quit'
RadialBlur	'RdlB'
Rasterize	'Rstr'
RasterizeTypeSheet	'RstT'
RemoveBlackMatte	'RmvB'
RemoveLayerMask	'RmvL'
RemoveWhiteMatte	'RmvW'
Rename	'Rnm '
ReplaceColor	'RplC'
Reset	'Rset'
ResizeImage	"1333cf0c-cb67-11d1-bc43-0060b0a13dc4"
Reticulation	'Rtcl'
Revert	'Rvrt'
Ripple	'Rple'
Rotate	'Rtte'
RoughPastels	'RghP'
Save	'save'
Select	'slct'
SelectiveColor	'SlcC'
Set	'setd'
SharpenEdges	'ShrE'
Sharpen	'Shrp'
SharpenMore	'ShrM'
Shear	'Shr '
Show	'Shw '
Similar	'Smlr'
SmartBlur	'SmrB'

Event	4-char ID or String
Smooth	'Smth'
SmudgeStick	'SmdS'
Solarize	'Slrz'
Spatter	'Spt '
Spherize	'Sphr'
SplitChannels	'SplC'
Sponge	'Spng'
SprayedStrokes	'SprS'
StainedGlass	'StnG'
Stamp	'Stmp'
Stop	'Stop'
Stroke	'Strk'
Subtract	'Sbtr'
SubtractFrom	'SbtF'
Sumie	'Smie'
TakeMergedSnapshot	'TkMr'
TakeSnapshot	'TkSn'
TextureFill	'TxtF'
Texturizer	'Txtz'
Threshold	'Thrs'
Tiles	'Tls '
TornEdges	'TrnE'
TraceContour	'TrcC'
Transform	'Trnf'
Trap	'Trap'
Twirl	'Twrl'
Underpainting	'Undr'
Undo	'undo'
Ungroup	'Ungr'
Unlink	'Unlk'
UnsharpMask	'UnsM'
Variations	'Vrtn'
Wait	'Wait'

Event	4-char ID or String
WaterPaper	'WtrP'
Watercolor	'Wtrc'
Wave	'Wave'
Wind	'Wnd '
ZigZag	'ZgZg'
BackLight	'BacL'
FillFlash	'File'
ColorCast	'ColE'



# Index

## A

- Action Manager 194
- actions
  - command lists 40
  - descriptions 43
  - descriptors 37
  - playing 47
- active document 45
- activePrinter 97
- Add Noise filter
- adjustments
  - brightness 56
  - color 198
  - color balance 56, 62
  - contrast 56, 60
  - curves 56
  - highlights 63
  - levels 56, 60
  - shadows 63
  - temperature 62
- Adobe Illustrator, exporting paths to 106
- alpha channels
  - defined 73
  - from transparency (TIFF documents) 187
  - opacity 73
  - saving
    - in BMP documents 70
    - in PDF documents 149
    - in PICT documents 154
    - in PICT resources 155
    - in Pixar documents 157
    - in PSD documents 153
    - in RAW documents 165
    - in SGIRGB documents 172
    - in Targa documents 177
    - in TIFF documents 187
- anchor points
  - adding 144
- annotations, importing 93
- anti aliasing
  - text 180
- application
  - activating 47
  - checking if feature enabled 49
  - defaults 159
  - location 46
  - preferences 159
- artLayers, *See* layers
- Asian text 161
- authors 100
- auto kerning 180, 198
- auto leading 185
- auto spacing, contact sheets 83

- available memory 46
- Average filter 56

## B

- background color
  - application 45
  - galleries 111
- background layers 54
- backgroundColor 97
- baseline shift 180
- batch command 47
- batches
  - destination folder 67, 198
  - specifying options 67
- beeping 159
- bitmap documents
  - converting to 199
  - depth type 199
  - halftone type 199
  - opening 208
  - saving 70
- bitmap images
  - See* bitmap documents
- black and white images 63
- bleedWidth 97
- blending modes
  - layer sets 130
  - layers 54
- Blur filter 56
- blur filters
  - Average 56
  - Blur More 56
  - Gaussian Blur 57
  - Lens Blur 58
  - Motion Blur 59
  - Radial Blur 59
  - Smart Blur 60
- Blur More filter 56
- BMP documents
  - See* bitmap documents
- brightness 56
  - adjusting 56
  - equalizing 61
- build 45

## C

- caches
  - images 160
  - purging 50
- camera raw documents
  - opening 71
  - settings 200
  - size options 200

- canvas
    - flipping 93
    - resizing 93
  - canvas, defined 89
  - caption 97
  - captions
    - contact sheets 83
    - documents 100
    - gallery images 112
    - gallery thumbnails 116
    - images 100
  - centerCropMarks 97
  - changeProgressText 47
  - channels
    - activating 89
    - adding 75
    - adjusting 56
    - alpha *See* alpha channels
    - creating 75
    - deleting 73
    - displaying in color 159
    - duplicating 73
    - making visible 73
    - merging 73
    - mixing 62
    - splitting 94
    - spot *See* spot channels
    - types of 73
  - clipping paths
    - from paths 140
    - from text 186
  - Clouds filter 56
  - CMYKColor 80
  - color balance, adjusting 62
  - color picker 159
  - color profiles
    - changing 92
    - determining type of 90
    - naming 89
  - color profiles, *see* individual document formats
  - color samplers
    - adding 82
    - creating 82
    - moving 81
    - removing 81
  - colorBars 97
  - colorHandling 97
  - colors
    - active links 111
    - adjusting 198
    - balancing 56
    - channels 73
    - CMYK 80
    - custom settings 114
    - in galleries 111
    - inverting 61
    - modifying 63
    - none 136
    - preserving (GIF only) 117
    - reduction 107
    - settings 45
    - solid color objects 173
    - testing if equal 173
    - visited links 111
  - comments, layer comps 127
  - compatibility, maximizing 160
  - component channels
    - color balance 56
    - defined 73
    - listing 89
    - See* composite channels
  - composite channels 73
    - See* component channels
  - Compuserve GIF documents
    - opening 208
    - saving 117
  - contact sheets
    - captions 83
    - columns 83
    - dimensions 83
    - making 49
    - rows 83
  - contrast
    - adjusting 56
    - adjusting automatically 60
    - camera raw settings 71
    - midtones 63
  - copies 97
  - copyrights 100
  - cornerCropMarks 97
  - count items
    - adding 85
    - creating 85
    - removing 84
  - cropping 92
  - current tool name 45
  - cursors 161
  - curves, adjusting 56
  - Custom filters 56
- D**
- DCS 1 documents, saving 86
  - De-Interlace filter 57
  - desaturate 61
  - Despeckle filter 57
  - dialogs
    - displaying 45
  - Difference Clouds filter 57
  - Diffuse Glow filter 57
  - Displace filter 57
  - distort filters
    - Diffuse Glow 57
    - Displace 57
    - Glass Effect 57
    - Ocean Ripple 59
    - Pinch 59
    - Polar Coordinates 59
    - Ripple 59
    - Shear 60

- Spherize 60
- Twirl 60
- Wave 60
- Zigzag 60
- doAction 47
- document formats, *see* individual document formats
- DocumentPrintSettings 97
- documents 89
  - activating 45
  - adding 103
  - closing 91
  - code sample 95
  - color profiles 89
  - color samplers 89
  - counting items 90
  - counting objects 92
  - cropping 36, 92
  - dimensions 89
  - duplicating 36, 92
  - exporting 93
  - info 100
  - loading 49
  - managed 90
  - measurement scale 90
  - metadata 90, 100
  - open with Photoshop dialog 50
  - opening 50
  - optimizing for web 107
  - printing 93
  - resizing 93
  - resolution 91
  - saving 93, 94
  - suspending history 94
  - trapping (CMYK) 94
  - trimming 95
- doForcedProgress 47
- doProgress 48
- doProgressSegmentTask 48
- doProgressTask 48
- Dust and Scratches filter 57

## E

- Enable Info
  - constants 192
  - grammar 191
  - operator precedence 192
  - variables 193
- EPSSaveOptions 105
- equalize 61
- event IDs
  - using ScriptListener to find 217
- Events Manager 46
- executing scripts 33
- exif 100
- exporting
  - documents 93
  - paths 106
  - to Illustrator 106
  - to Web 107

- ExportOptionsIllustrator 106
- ExportOptionsSaveForWeb 107

## F

- file extensions
  - format 162
  - including 159
  - script files 32
- file metadata 100
- files
  - merging 49
- filetypes
  - macOS 46
  - Windows 47
- filling
  - paths 139
  - selections 168
- filter, *see* individual filter names
- flip 97
- Folder object 32
- fonts
  - detecting 45
  - determining family of 178
  - determining style of 178
- formats, *see* individual document formats

## G

- galleries 114
  - background color 111
  - banners 110
  - captions 112
  - color options 111
  - credits 112
  - dimensions 112
  - filenames 112
  - link colors 111
  - making 47, 49
  - metadata 114
  - photographer 110
  - security text 115
  - thumbnail images 116
- GalleryBannerOptions 110
- GalleryCustomColorOptions 111
- GalleryImagesOptions 112, 113
- GalleryOptions 114
- GallerySecurityOptions 115
- GalleryThumbnailOptions 116
- Gaussian Blur filter 57
- GIF documents
  - See* Compuserve GIF documents
- GIFSaveOptions 117
- Glass Effect filter 57
- glyph scaling 180–183
- GrayColor 118
- grids 160
- grouped layers 54
- guides 160

**H**

- halftone screen 69
- hanging punctuation 181
- hardProof 97
- High Pass filter 57
- highlights
  - adjusting 63
  - color balance 56
- histograms
  - channels 73
- history log 162
- history states
  - activating 89
  - allowing nonlinear 160
  - default number of 161
  - snapshot 121
  - suspending 94
- HSBColor 123
- hyphenation 182

**I**

- IDs
  - getting 37
  - PICT Resource 155
  - property 43
  - runtime 47
  - runtime to string 51
  - string to runtime 51
  - string to type 51
  - type to char 51
- Illustrator
  - See Adobe Illustrator
- image
  - resizing 94
- image pyramids 187
- images
  - bitmap 69
  - black and white 63
  - caches 160
  - captions 112
  - definition of 89
  - desaturating 61
  - equalizing 61
  - filetypes 46
  - from split channels 94
  - inverting colors 61
  - previewing 160
  - pyramids 187
  - resizing 93
  - resizing in galleries 113
  - thumbnails 116
- indexed color model 124
- IndexedConversionOptions 124
- individual document formats, examples 94
- installing scripts 33
- Intent 97
- interpolate 97
- interpolation 160

**J**

- JavaScript
  - supported features 32
- JavaScript Resource
  - Enable Info grammar 191
  - javascriptresource syntax 190
- javascriptresource tag 190
- JPEG
  - quality 125
- JPEG documents
  - quality 125
  - saving 125
- JPEG options
  - scans 125
- JPEGSaveOptions 125
- justification 182

**K**

- Kerning 180
  - text
    - auto kerning 198
- keyboard behavior 160

**L**

- LabColor 126
- labels 97
- languages 182
- layer comps 127
  - adding 128
  - applying 127
  - in documents 90
- layer sets
  - adding 132
  - art layers in 130
  - duplicating 131
  - in documents 90
  - linked layers in 130
  - linking 131
  - locking contents 130
  - moving 131
  - nesting 130
  - opacity 130
  - unlinking 131
- layer styles, applying 60
- LayerComps 128
- layered TIFFdocuments, saving 159
- Layers 129
- layers
  - adding 66
  - applying styles 60
  - background 54
  - blending mode 54
  - bounds 54
  - clipboard commands 60
  - comps 127
  - copying 60
  - duplicating 61
  - flattening 93
  - grouping 54

- in documents 90
- inverting 61
- kind 54
- linking 61
- locking contents 54–55
- making visible 55
- merging 61
- merging visible 93
- moving 62
- rasterizing 93
- rasterizing contents 62
- removing 66
- resizing 63
- rotating 63
- saving in PDF documents 149
- unlinking 63
- LayerSet 130
- LayerSets 132
- layersets
  - merging 131
- leading 182, 185
- Lens Blur filter
  - applying 58
- Lens Flare filter 58
- letter spacing 181–184
- levels
  - adjusting 56
  - adjusting automatically 56
- ligatures 180–182
- linked layers 61
  - unlinking 63
- links
  - colors 111

## M

- MacOS
  - filetypes 46
- managed documents 90
- mapBlack 97
- maximizing compatibility 160
- Maximum filter 58
- Median Noise filter 58
- memory 46
- merging
  - layers 61
  - visible layers 93
- metadata
  - document 90
  - document object 100
  - galleries 114
  - xmp 91, 189
- methods
  - batch 47
- midtones
  - color balance 56
- Minimum filter 59
- Motion Blur filter 59

## N

- negative 97
- noise filters
  - Add Noise
  - Despeckle 57
  - Dust and Scratches 57
  - Median Noise 58
- nonlinear history 160
- notifications
  - events within scripts 137
- notifiers
  - adding 138
  - event IDs 217
  - removing 137
- NTSC filter 59

## O

- Ocean Ripple filter 59
- Offset filter 59
- old style type 184
- opacity
  - channels 73
  - gallery security text 115
  - layer fill 54
  - layer sets 130
  - layers 55
  - picture packages 156
- open options
  - DICOM format 88
  - EPS format 104
  - PDF format 148
  - Photo CD format 152
  - RAW format 71, 164
- optimizing 107
- other filters
  - Custom 56
  - High Pass 57
  - Maximum 58
  - Minimum 59
  - Offset 59

## P

- palettes 161
- pasting 93
- path 46
- path items
  - adding 143
  - deselecting 139
  - filling 139
  - from text 186
  - making selection 140
  - path points 175
  - selecting 140
  - specifying path kind 139
  - stroking 140
  - sub items 139
  - sub path info 174
  - sub path items 175
  - work path from selection 168

- path point info
  - anchor points 145
  - left direction 145
  - right direction 145
- path points
  - anchor points 144
  - left direction 144
  - right direction 144
- PathItems 143
- paths
  - See *path items*
- PDF documents
  - opening 148
  - saving 149
- PDF presentations
  - auto advance 163
  - making 49
  - output format 163
  - transition type 163
- Photo CD discs, opening 152
- photo filtering 62
- photo galleries
  - See galleries
- photomerge 49
- Photoshop documents
  - opening 208
  - saving 153
- Photoshop files, maximizing compatibility 160
- PICT documents
  - opening 208
  - saving 154
- PICT resources
  - opening 208
  - saving 155
- picture packages
  - contents 156
  - flattening 156
  - making 49
  - opacity 156
  - options 156
  - text properties 156
- Pinch filter 59
- Pixar documents
  - opening 208
  - saving 157
- PixarSaveOptions 157
- pixels
  - aspect ratio 90
  - doubling 161
  - equalizing 61
  - interpolation 160
  - locking 55
  - unit measures 215
- playback options 46
- playbackDisplayDialogs 46
- plug-in folder
  - additional plug-in folder 162
- PNG 8 documents, saving 107
- PNG documents
  - saving 158

- PNGSaveOptions 158
- Polar Coordinates filter 59
- posterrize 62
- postscript encoding 93
- PostScript names 178
- posX 97
- posY 97
- Preferences 159
- PresentationOptions 163
- presentations
  - making 49
  - PDF presentations
- printBorder 97
- printing, documents 93
- printOneCopy 93
- printSelected 97
- printSettings 90
- printSpace 97
- property
  - measurementLog 46
- PSD documents
  - opening 208
  - saving 153
- purging 50

## Q

- quickMaskMode 91
- quote style 161

## R

- Radial Blur filter 59
- rasterize 62
- rasterizing
  - document layers 93
- RAW documents
  - opening 164
- RawSaveOptions 165
- recentFiles 46
- registrationMarks 97
- render filters
  - Clouds 56
  - Difference Clouds 57
  - Lens Flare 58
- renderIntent 97
- resolution
  - bitmap conversions 69
  - documents 91
- RGBColor 166
- Ripple filter 59
- rotation 63
- ruler units 161
- runMenuItem 50

## S

- save as 94
- saved 91
- saving 93
- saving, *see* individual document formats.

- scale 98
- scripting interface
  - build date 46
  - version 46
- scriptingVersion 46
- scripts
  - automation 190, 194
  - enabling/disabling in menu 190
  - executing 33
  - grouping in menu 190
  - installing 33
  - startup 33
  - terminology dictionary 194
  - valid file extensions 32
- Scripts Events Manager 46
- selected areas 91
- selections 167
  - boundaries 167
  - clearing 167
  - copying 167
  - cutting 167
  - deselecting 167
  - feathering 167
  - filling 168
  - from paths 140
  - making work path from 168
  - resizing 167, 168
  - rotating 168
  - smoothing 169
  - stroking 169
- selective color 63
- SGIRGB documents
  - saving 172, 212
- SGIRGBSaveOptions 172, 198
- shadows
  - adjusting 63
  - color balance 56
- Sharpen Edges filter 59
- Sharpen filter 59
- sharpen filters
  - Sharpen 59
  - Sharpen Edges 59
  - Sharpen More 59
  - Unsharp Mask 60
- Sharpen More filter 59
- Shear filter 60
- Smart Blur filter 60
- smart quotes 161
- Spherize filter 60
- spot channels
  - defined 73
  - merging into component channels 74
  - opacity 73
  - saving
    - in DCS 2 documents 87
    - in PDF documents 150
    - in PSD documents 153
    - in RAW documents 165
    - in SGIRGB documents 172
    - in TIFF documents 187

- spotColors 172
- startup scripts 33
- strike thru 185
- stroking
  - default stroke color 45
  - path items 140
  - selections 169
- styles, applying 60
- sub path items 139
- systemInformation 46

## T

- temperature 62
- terminology dictionary
  - defined 194
  - syntax 194
- text
  - Asian 161
  - auto kerning 180
  - auto leading 185
  - captions 112
  - color
  - composer 185
  - content 180
  - creating paths from 186
  - formatting 185
  - gallery security 115, 205
  - hyphenation 182
  - in picture packages 156
  - justification 182
  - languages 182
  - offset 180
  - orientation 181
  - spacing 181–184
  - tracking 185
  - wrapping 182
- text composer 185
- text fonts
  - See fonts
- text items
  - See text
- text layers
  - adding contents 180
  - creating 54
- Texture Fill filter 60
- texture filters, Texture Fill 60
- threshold 63
- thumbnails 116
  - Mac OS 160
  - Windows 162
- TIFF documents
  - layered 159
  - saving 187
- togglePalettes 51
- tool tips 161
- toolSupportsBrushes 51
- tracking, text 185
- transmission info 100
- trapping 94

Twirl filter 60  
type units 162

## U

underlining 185  
units  
    ruler 161  
    type 162  
UnitValue object 32, 188  
Unsharp Mask filter 60  
URLs, document 100  
UTF8 Encoding 114

## V

vectorData 98  
version  
    application 47  
    scripting interface 46  
video alpha 162  
video filters  
    De-Interlace 57  
    NTSC 59  
visibility  
    channels 73  
    layer comps 127

layers 55

## W

warp 185  
Wave filter 60  
Web photo galleries  
    *See galleries.*  
webSnap 108  
width 91  
Windows  
    filetypes 47  
word spacing 181–184  
work paths  
    designating 209  
    from selected area 168  
wrapping, text 182

## X

XML 189  
xmp metadata 91, 189

## Z

Zigzag filter 60  
zoom 160