
NSWindowScripting Protocol Reference

[Cocoa > Scripting & Automation](#)



2006-05-23



Apple Inc.
© 2006 Apple Computer, Inc.
All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, mechanical, electronic, photocopying, recording, or otherwise, without prior written permission of Apple Inc., with the following exceptions: Any person is hereby authorized to store documentation on a single computer for personal use only and to print copies of documentation for personal use provided that the documentation contains Apple's copyright notice.

The Apple logo is a trademark of Apple Inc.

Use of the "keyboard" Apple logo (Option-Shift-K) for commercial purposes without the prior written consent of Apple may constitute trademark infringement and unfair competition in violation of federal and state laws.

No licenses, express or implied, are granted with respect to any of the technology described in this document. Apple retains all intellectual property rights associated with the technology described in this document. This document is intended to assist application developers to develop applications only for Apple-labeled computers.

Every effort has been made to ensure that the information in this document is accurate. Apple is not responsible for typographical errors.

Apple Inc.
1 Infinite Loop
Cupertino, CA 95014
408-996-1010

Apple, the Apple logo, AppleScript, Cocoa, Mac, and Mac OS are trademarks of Apple Inc., registered in the United States and other countries.

Simultaneously published in the United States and Canada.

Even though Apple has reviewed this document, APPLE MAKES NO WARRANTY OR REPRESENTATION, EITHER EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT, ITS QUALITY, ACCURACY, MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE. AS A RESULT, THIS DOCUMENT IS PROVIDED "AS IS," AND YOU, THE READER, ARE ASSUMING THE ENTIRE RISK AS TO ITS QUALITY AND ACCURACY.

IN NO EVENT WILL APPLE BE LIABLE FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES RESULTING FROM ANY

DEFECT OR INACCURACY IN THIS DOCUMENT, even if advised of the possibility of such damages.

THE WARRANTY AND REMEDIES SET FORTH ABOVE ARE EXCLUSIVE AND IN LIEU OF ALL OTHERS, ORAL OR WRITTEN, EXPRESS OR IMPLIED. No Apple dealer, agent, or employee is authorized to make any modification, extension, or addition to this warranty.

Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you. This warranty gives you specific legal rights, and you may also have other rights which vary from state to state.

Contents

NSWindowScripting Protocol Reference 5

Overview 5

Tasks 5

 Getting Scripting Attribute Information About a Window 5

 Setting Scripting Attribute Information for a Window 5

 Handling Script Commands 6

 Working with Ordered Indices 6

Instance Methods 6

 handleCloseScriptCommand: 6

 handlePrintScriptCommand: 7

 handleSaveScriptCommand: 7

 hasCloseBox 7

 hasTitleBar 8

 isFloatingPanel 8

 isMiniaturizable 8

 isModalPanel 8

 isResizable 8

 isZoomable 9

 orderedIndex 9

 setIsMiniaturized: 9

 setIsVisible: 10

 setIsZoomed: 10

 setOrderedIndex: 10

Document Revision History 11

Index 13

NSWindowScripting Protocol Reference

(informal protocol)

Framework	/System/Library/Frameworks/AppKit.framework
Companion guide	Cocoa Scripting Guide
Declared in	NSWindowScripting.h

Overview

Category on `NSWindow`. Supports window scripting for all scriptable Cocoa applications by getting and setting standard properties and handling the `close`, `print`, and `save` AppleScript commands.

Tasks

Getting Scripting Attribute Information About a Window

- [hasCloseBox](#) (page 7)
Returns YES if the receiver has a close box.
- [hasTitleBar](#) (page 8)
Returns YES if the receiver has a title bar.
- [isFloatingPanel](#) (page 8)
Returns YES if the receiver is a floating panel.
- [isMiniaturizable](#) (page 8)
Returns YES if the receiver can be miniaturized (has a minimize button).
- [isModalPanel](#) (page 8)
Returns YES if the receiver is an application-modal panel.
- [isResizable](#) (page 8)
Returns YES if the receiver is resizable (has a size control).
- [isZoomable](#) (page 9)
Returns YES if the receiver is zoomable (has a zoom button).

Setting Scripting Attribute Information for a Window

- [setIsMiniaturized:](#) (page 9)
Sets the receiver's miniaturized state to the value specified by *flag*.

- `setVisible:` (page 10)
Sets the receiver's visible state to the value specified by *flag*.
- `setZoomed:` (page 10)
Sets the receiver's zoomed state to the value specified by *flag*.

Handling Script Commands

- `handleCloseScriptCommand:` (page 6)
Handles the `close` AppleScript command by attempting to close the window (and its associated document, if any).
- `handlePrintScriptCommand:` (page 7)
Handles the `print` AppleScript command by attempting to print the contents of the window (or its associated document, if any).
- `handleSaveScriptCommand:` (page 7)
Handles the `save` AppleScript command by attempting to save the window (and its associated document, if any).

Working with Ordered Indices

- `orderedIndex` (page 9)
Returns the zero-based position of the receiver based on its order from front to back among all application windows.
- `setOrderedIndex:` (page 10)
Sets the zero-based position of the receiver, based on its order from front to back among all visible application windows, to the value specified by *index*. If *index* is out of range, sets the position to the nearest value that is in range.

Instance Methods

handleCloseScriptCommand:

Handles the `close` AppleScript command by attempting to close the window (and its associated document, if any).

```
- (id)handleCloseScriptCommand:(NSCloseCommand *)command
```

Discussion

Extracts `close` command arguments from the *command* object and uses them to determine how to close the associated document—specifically, whether to ignore unsaved changes, save changes automatically, or ask the user and to identify the file in which to save the document (by default, the file that was opened or previously saved to, or an “untitled” name if the file has never been saved).

If there is a corresponding document and the window is the main window of the document, it forwards the `close` command to the corresponding document; otherwise, the window sends itself a `performClose` message, if it has a close box. This may have been handled differently in versions of Mac OS X prior to version 10.3.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

handlePrintScriptCommand:

Handles the `print` AppleScript command by attempting to print the contents of the window (or its associated document, if any).

```
- (id)handlePrintScriptCommand:(NSScriptCommand *)command
```

Discussion

If there is a corresponding document and the window is the main window of the document, it forwards the `print` command to the corresponding document; otherwise, the window sends itself a `print` message. This may have been handled differently in versions of Mac OS X prior to version 10.3.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

handleSaveScriptCommand:

Handles the `save` AppleScript command by attempting to save the window (and its associated document, if any).

```
- (id)handleSaveScriptCommand:(NSScriptCommand *)command
```

Discussion

The default version invokes the same named method of the window's document, if the window is the one being saved.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

hasCloseBox

Returns YES if the receiver has a close box.

```
- (BOOL)hasCloseBox
```

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

hasTitleBar

Returns YES if the receiver has a title bar.

- (BOOL)hasTitleBar

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

isFloatingPanel

Returns YES if the receiver is a floating panel.

- (BOOL)isFloatingPanel

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

isMiniaturizable

Returns YES if the receiver can be miniaturized (has a minimize button).

- (BOOL)isMiniaturizable

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

isModalPanel

Returns YES if the receiver is an application-modal panel.

- (BOOL)isModalPanel

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

isResizable

Returns YES if the receiver is resizable (has a size control).

- (BOOL)isResizable

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

isZoomable

Returns YES if the receiver is zoomable (has a zoom button).

- (BOOL)isZoomable

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

orderedIndex

Returns the zero-based position of the receiver based on its order from front to back among all application windows.

- (int)orderedIndex

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

setIsMiniaturized:

Sets the receiver's miniaturized state to the value specified by *flag*.

- (void)setIsMiniaturized:(BOOL)*flag*

Discussion

Depending on the current miniaturized state and the value of *flag*, the window may be minimized to the Dock or expanded from the Dock.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

setIsVisible:

Sets the receiver's visible state to the value specified by *flag*.

- (void)setIsVisible:(BOOL)*flag*

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

setIsZoomed:

Sets the receiver's zoomed state to the value specified by *flag*.

- (void)setIsZoomed:(BOOL)*flag*

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

setOrderedIndex:

Sets the zero-based position of the receiver, based on its order from front to back among all visible application windows, to the value specified by *index*. If *index* is out of range, sets the position to the nearest value that is in range.

- (void)setOrderedIndex:(int)*index*

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSWindowScripting.h

Document Revision History

This table describes the changes to *NSWindowScripting Protocol Reference*.

Date	Notes
2006-05-23	First publication of this content as a separate document.

REVISION HISTORY

Document Revision History

Index

H

handleCloseScriptCommand: <NSWindow> instance method [6](#)
handlePrintScriptCommand: <NSWindow> instance method [7](#)
handleSaveScriptCommand: <NSWindow> instance method [7](#)
hasCloseBox <NSWindow> instance method [7](#)
hasTitleBar <NSWindow> instance method [8](#)

I

isFloatingPanel <NSWindow> instance method [8](#)
isMiniaturizable <NSWindow> instance method [8](#)
isModalPanel <NSWindow> instance method [8](#)
isResizable <NSWindow> instance method [8](#)
isZoomable <NSWindow> instance method [9](#)

O

orderedIndex <NSWindow> instance method [9](#)

S

setIsMiniaturized: <NSWindow> instance method [9](#)
setIsVisible: <NSWindow> instance method [10](#)
setIsZoomed: <NSWindow> instance method [10](#)
setOrderedIndex: <NSWindow> instance method [10](#)