

---

# NSPopUpButton Class Reference

User Experience: Menus



2008-10-15



Apple Inc.  
© 2008 Apple 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, Cocoa, Mac, Mac OS, and Quartz 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

## **NSPopupMenu Class Reference** 5

---

Class at a Glance	5
Overview	6
Tasks	6
Initializing an NSPopupMenu	6
Setting the Type of Menu	6
Inserting and Deleting Items	6
Getting the User's Selection	7
Setting the Current Selection	7
Getting Menu Items	7
Getting the Indices of Menu Items	8
Setting the Cell Edge to Pop out in Restricted Situations	8
Setting the Title	8
Setting the Image	8
Setting the State	9
Instance Methods	9
addItemWithTitle:	9
addItemWithTitle:	9
autoenablesItems	10
indexOfItem:	11
indexOfItemWithRepresentedObject:	11
indexOfItemWithTag:	11
indexOfItemWithTarget:andAction:	12
indexOfItemWithTitle:	12
indexOfSelectedItem	13
initWithFrame:pullsDown:	13
insertItemWithTitle:atIndex:	14
itemArray	15
itemAtIndex:	15
itemTitleAtIndex:	16
itemTitles	16
itemWithTitle:	17
lastItem	17
menu	18
numberOfItems	18
objectValue	18
preferredEdge	19
pullsDown	19
removeAllItems	19
removeItemAtIndex:	20
removeItemWithTitle:	21

- selectedItem 21
- selectItem: 22
- selectItemAtIndex: 22
- selectItemWithTag: 22
- selectItemWithTitle: 23
- setAutoenablesItems: 24
- setImage: 24
- setMenu: 24
- setObjectValue: 25
- setPreferredEdge: 25
- setPullsDown: 25
- setTitle: 26
- synchronizeTitleAndSelectedItem 26
- titleOfSelectedItem 27
- Notifications 27
  - NSPopUpButtonWillPopUpNotification 27

---

**Document Revision History 29**

---

**Index 31**

---

# NSPopUpButton Class Reference

---

<b>Inherits from</b>	NSButton : NSControl : NSView : NSResponder : NSObject
<b>Conforms to</b>	NSUserInterfaceValidations (NSButton) NSAnimatablePropertyContainer (NSView) NSCoding (NSResponder) NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/AppKit.framework
<b>Availability</b>	Available in Mac OS X v10.0 and later.
<b>Companion guide</b>	Application Menu and Pop-up List Programming Topics for Cocoa
<b>Declared in</b>	NSPopUpButton.h
<b>Related sample code</b>	ColorMatching FinalCutPro_AppleEvents MenuMadness Quartz Composer WWDC 2005 TextEdit WhackedTV

## Class at a Glance

An `NSPopUpButton` object controls a pop-up menu or a pull-down menu from which a user can select an item.

## Principal Attributes

---

- An `NSMenu`

### Interface Builder

Use Interface Builder to add a pop-up or pull-down menu to a window or panel.

## Commonly Used Methods

---

[selectedItem](#) (page 21)

Returns the currently selected item.

[indexOfSelectedItem](#) (page 13)

Returns an integer identifying the currently selected item.

[titleOfSelectedItem](#) (page 27)

Returns a string identifying the currently selected item.

## Overview

The `NSPopUpButton` class defines objects that implement the pop-up and pull-down menus of the graphical user interface.

An `NSPopUpButton` object uses an `NSPopUpButtonCell` object to implement its user interface.

Note that while a menu is tracking, adding, removing, or changing items on the menu is not reflected.

## Tasks

### Initializing an NSPopUpButton

- [initWithFrame:pullsDown:](#) (page 13)

Returns an `NSPopUpButton` object initialized to the specified dimensions.

### Setting the Type of Menu

- [setPullsDown:](#) (page 25)

Sets whether the receiver behaves as a pull-down or pop-up menu.

- [pullsDown](#) (page 19)

Returns a Boolean value indicating the behavior of the control's menu.

- [setAutoenablesItems:](#) (page 24)

Sets whether the receiver automatically enables and disables its items every time a user event occurs.

- [autoenablesItems](#) (page 10)

Returns whether the receiver automatically enables and disables its items every time a user event occurs.

### Inserting and Deleting Items

- [addItemWithTitle:](#) (page 9)

Adds an item with the specified title to the end of the menu.

- [addItemsWithTitles:](#) (page 9)

Adds multiple items to the end of the menu.

- [insertItemWithTitle:atIndex:](#) (page 14)

Inserts an item at the specified position in the menu.

- [removeAllItems](#) (page 19)  
Removes all items in the receiver's item menu.
- [removeItemWithTitle:](#) (page 21)  
Removes the item with the specified title from the menu.
- [removeItemAtIndex:](#) (page 20)  
Removes the item at the specified index.

## Getting the User's Selection

- [selectedItem](#) (page 21)  
Returns the menu item last selected by the user.
- [titleOfSelectedItem](#) (page 27)  
Returns the title of the item last selected by the user.
- [indexOfSelectedItem](#) (page 13)  
Returns the index of the item last selected by the user.
- [objectValue](#) (page 18)  
Returns the index of the selected item.

## Setting the Current Selection

- [selectItem:](#) (page 22)  
Selects the specified menu item.
- [selectItemAtIndex:](#) (page 22)  
Selects the item in the menu at the specified index.
- [selectItemWithTag:](#) (page 22)  
Selects the menu item with the specified tag.
- [selectItemWithTitle:](#) (page 23)  
Selects the item with the specified title.
- [setObjectValue:](#) (page 25)  
Selects the item at a specific index using an object value.

## Getting Menu Items

- [menu](#) (page 18)  
Returns the pop-up button's associated menu.
- [setMenu:](#) (page 24)  
Sets the pop-up button's associated menu .
- [numberOfItems](#) (page 18)  
Returns the number of items in the menu.
- [itemArray](#) (page 15)  
Returns the items in the menu.
- [itemAtIndex:](#) (page 15)  
Returns the menu item at the specified index.

- [itemTitleAtIndex:](#) (page 16)  
Returns the title of the item at the specified index.
- [itemTitles](#) (page 16)  
Returns the titles of all of the items in the menu.
- [itemWithTitle:](#) (page 17)  
Returns the menu item with the specified title.
- [lastItem](#) (page 17)  
Returns the last item in the menu.

## Getting the Indices of Menu Items

- [indexOfItem:](#) (page 11)  
Returns the index of the specified menu item.
- [indexOfItemWithTag:](#) (page 11)  
Returns the index of the menu item with the specified tag.
- [indexOfItemWithTitle:](#) (page 12)  
Returns the index of the item with the specified title.
- [indexOfItemWithRepresentedObject:](#) (page 11)  
Returns the index of the menu item that holds the specified represented object.
- [indexOfItemWithTarget:andAction:](#) (page 12)  
Returns the index of the menu item with the specified target and action.

## Setting the Cell Edge to Pop out in Restricted Situations

- [preferredEdge](#) (page 19)  
Returns the edge of the receiver next to which the pop-up menu is displayed under restrictive screen conditions.
- [setPreferredEdge:](#) (page 25)  
Sets the edge of the receiver next to which the pop-up menu should appear under restrictive screen conditions.

## Setting the Title

- [setTitle:](#) (page 26)  
Sets the string displayed in the receiver when the user isn't pressing the mouse button.

## Setting the Image

- [setImage:](#) (page 24)  
This method has no effect.

## Setting the State

- [synchronizeTitleAndSelectedItem](#) (page 26)  
Ensures that the item being displayed by the receiver agrees with the selected item.

## Instance Methods

### addItemWithTitle:

Adds multiple items to the end of the menu.

```
- (void)addItemWithTitle:(NSArray *)itemTitles
```

#### Parameters

*itemTitles*

An array of `NSString` objects containing the titles of the items you want to add. Each string in the array should be unique. If an item with the same title already exists in the menu, the existing item is removed and the new one is added.

#### Discussion

If you want to move an item, it's better to invoke [removeItemWithTitle:](#) (page 21) explicitly and then send this method. After adding the items, this method uses the [synchronizeTitleAndSelectedItem](#) (page 26) method to make sure the item being displayed matches the currently selected item.

Since this method searches for duplicate items, it should not be used if you are adding items to an already populated menu with more than a few hundred items. Add items directly to the receiver's menu instead.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

- [insertItemWithTitle:atIndex:](#) (page 14)
- [removeAllItems](#) (page 19)
- [removeItemWithTitle:](#) (page 21)

#### Related Sample Code

AlbumToSlideshow  
MovieAssembler  
TimelineToTC

#### Declared In

NSPopupMenu.h

### addItemWithTitle:

Adds an item with the specified title to the end of the menu.

```
- (void)addItemWithTitle:(NSString *)title
```

**Parameters***title*

The title of the menu-item entry. If an item with the same title already exists in the menu, the existing item is removed and the new one is added.

**Discussion**

If you want to move an item, it's better to invoke [removeItemWithTitle:](#) (page 21) explicitly and then send this method. After adding the item, this method calls the [synchronizeTitleAndSelectedItem](#) (page 26) method to make sure the item being displayed matches the currently selected item.

Since this method searches for duplicate items, it should not be used if you are adding an item to an already populated menu with more than a few hundred items. Add items directly to the receiver's menu instead.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [insertItemWithTitle:AtIndex:](#) (page 14)
- [removeItemWithTitle:](#) (page 21)
- [setTitle:](#) (page 26)

**Related Sample Code**

MovieAssembler  
 QTAudioContextInsert  
 Quartz Composer WWDC 2005 TextEdit  
 UIElementInspector  
 WhackedTV

**Declared In**

NSPopupMenu.h

**autoenablesItems**

Returns whether the receiver automatically enables and disables its items every time a user event occurs.

- (BOOL)autoenablesItems

**Return Value**

YES if the receiver automatically enables and disables items; otherwise, NO. The default value is YES.

**Discussion**

For more information on enabling and disabling menu items, see the NSMenuValidation protocol specification.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [setAutoenablesItems:](#) (page 24)

**Declared In**

NSPopupMenu.h

**indexOfItem:**

Returns the index of the specified menu item.

```
- (NSInteger)indexOfItem:(NSMenuItem *)anObject
```

**Parameters**

*anObject*

The menu item whose index you want.

**Return Value**

The index of the item or -1 if no such item was found.

**Discussion**

This method invokes the method of the same name of its `NSPopUpButtonCell` object.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

`NSPopUpButton.h`

**indexOfItemWithRepresentedObject:**

Returns the index of the menu item that holds the specified represented object.

```
- (NSInteger)indexOfItemWithRepresentedObject:(id)anObject
```

**Parameters**

*anObject*

The represented object associated with a menu item.

**Return Value**

The index of the menu item that owns the specified object, or -1 if no such menu item was found.

**Discussion**

Represented objects bear some direct relation to the title or image of a menu item; for example, an item entitled "100" might have an `NSNumber` object encapsulating that value as its represented object. This method invokes the method of the same name of its `NSPopUpButtonCell` object.

**Availability**

Available in Mac OS X v10.0 and later.

**Related Sample Code**

`OutputBinsPDE`

**Declared In**

`NSPopUpButton.h`

**indexOfItemWithTag:**

Returns the index of the menu item with the specified tag.

```
- (NSInteger)indexOfItemWithTag:(NSInteger)tag
```

**Parameters***tag*

The tag of the menu item you want.

**Return Value**

The index of the item or -1 if no item with the specified tag was found.

**Discussion**This method invokes the method of the same name of its `NSPopUpButtonCell` object.**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSPopUpButton.h

**indexOfItemWithTarget:andAction:**

Returns the index of the menu item with the specified target and action.

- (NSInteger)indexOfItemWithTarget:(id)target andAction:(SEL)actionSelector

**Parameters***target*

The target object associated with the menu item.

*actionSelector*

The action method associated with the menu item.

**Return Value**

The index of the menu item, or -1 if no menu item contains the specified target and action.

**Discussion**If you specify `NULL` for the *actionSelector* parameter, the index of the first menu item with the specified target is returned. This method invokes the method of the same name of its `NSPopUpButtonCell` object.**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSPopUpButton.h

**indexOfItemWithTitle:**

Returns the index of the item with the specified title.

- (NSInteger)indexOfItemWithTitle:(NSString \*)title

**Parameters***title*

The title of the item you want.

**Return Value**

The index of the item or -1 if no item with the specified title was found.

### Availability

Available in Mac OS X v10.0 and later.

### Related Sample Code

AlbumToSlideshow  
QTAudioContextInsert  
QTAudioExtractionPanel

### Declared In

NSPopUpButton.h

## indexOfSelectedItem

Returns the index of the item last selected by the user.

- (NSInteger)indexOfSelectedItem

### Return Value

The index of the selected item, or -1 if no item is selected.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- [selectedItem](#) (page 21)
- [titleOfSelectedItem](#) (page 27)

### Related Sample Code

AlbumToSlideshow  
GLUT  
iChatTheater  
QTAudioContextInsert  
UIElementInspector

### Declared In

NSPopUpButton.h

## initWithFrame:pullsDown:

Returns an NSPopUpButton object initialized to the specified dimensions.

- (id)initWithFrame:(NSRect) *frameRect* pullsDown:(BOOL) *flag*

### Parameters

*frameRect*

The frame rectangle for the button, specified in the parent view's coordinate system.

*flag*

YES if you want the receiver to display a pull-down menu; otherwise, NO if you want it to display a pop-up menu.

**Return Value**

An initialized `NSPopUpButton` object, or `nil` if the object could not be initialized.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [pullsDown](#) (page 19)
- [setPullsDown:](#) (page 25)

**Related Sample Code**

ButtonMadness  
 MenuItemView  
 MenuMadness  
 Quartz Composer WWDC 2005 TextEdit  
 Sketch+Accessibility

**Declared In**

`NSPopUpButton.h`

**insertItemWithTitle:atIndex:**

Inserts an item at the specified position in the menu.

```
- (void)insertItemWithTitle:(NSString *)title atIndex:(NSInteger)index
```

**Parameters**

*title*

The title of the new item. If an item with the same title already exists in the menu, the existing item is removed and the new one is added

*index*

The zero-based index at which to insert the item. Specifying 0 inserts the item at the top of the menu.

**Discussion**

If you want to move an item, it's better to invoke [removeItemWithTitle:](#) (page 21) explicitly and then send this method. After adding the item, this method uses the [synchronizeTitleAndSelectedItem](#) (page 26) method to make sure the item displayed matches the currently selected item.

Since this method searches for duplicate items, it should not be used if you are adding an item to an already populated menu with more than a few hundred items. Add items directly to the receiver's menu instead.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [addItemWithTitle:](#) (page 9)
- [addItemWithTitlees:](#) (page 9)
- [indexOfItemWithTitle:](#) (page 12)
- [removeItemWithTitle:](#) (page 21)

**Related Sample Code**

CAPlayThrough

QTAudioContextInsert  
QTAudioExtractionPanel

**Declared In**

NSPopUpButton.h

## itemArray

Returns the items in the menu.

- (NSArray \*)itemArray

**Return Value**

An array of id <NSMenuItem> objects representing the items in the menu.

**Discussion**

Usually you access the menu's items and modify the menu using the methods of NSPopUpButton rather than accessing the array of items directly.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [itemAtIndex:](#) (page 15)
- [insertItemWithTitle:atIndex:](#) (page 14)
- [removeItemAtIndex:](#) (page 20)

**Declared In**

NSPopUpButton.h

## itemAtIndex:

Returns the menu item at the specified index.

- (NSMenuItem \*)itemAtIndex:(NSInteger) *index*

**Parameters**

*index*

The index of the item you want.

**Return Value**

The menu item, or nil if no item exists at the specified index.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [itemWithTitle:](#) (page 17)
- [lastItem](#) (page 17)

**Related Sample Code**

QTAudioContextInsert  
QTAudioExtractionPanel

Quartz Composer WWDC 2005 TextEdit  
SampleScannerApp

**Declared In**

NSPopUpButton.h

**itemTitleAtIndex:**

Returns the title of the item at the specified index.

```
- (NSString *)itemTitleAtIndex:(NSInteger) index
```

**Parameters**

*index*

The index of the item you want.

**Return Value**

The title of the item, or an empty string if no item exists at the specified index.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [itemTitles](#) (page 16)

**Declared In**

NSPopUpButton.h

**itemTitles**

Returns the titles of all of the items in the menu.

```
- (NSArray *)itemTitles
```

**Return Value**

An array of `NSString` objects containing the titles of every item in the menu. The titles appear in the order in which the items appear in the menu.

**Discussion**

If the menu contains separator items, the array contains an empty string (@`''`) for each separator item.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [itemTitleAtIndex:](#) (page 16)

- [itemWithTitle:](#) (page 17)

- [numberOfItems](#) (page 18)

**Declared In**

NSPopUpButton.h

**itemWithTitle:**

Returns the menu item with the specified title.

```
- (NSMenuItem *)itemWithTitle:(NSString *)title
```

**Parameters**

*title*

The title of the menu item you want.

**Return Value**

The menu item, or `nil` if no item with the specified title exists in the menu.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [addItemWithTitle:](#) (page 9)
- [selectItemWithTitle:](#) (page 23)
- [itemAtIndex:](#) (page 15)
- [indexOfItemWithTitle:](#) (page 12)

**Declared In**

NSPopupMenu.h

**lastItem**

Returns the last item in the menu.

```
- (NSMenuItem *)lastItem
```

**Return Value**

The last menu item.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [itemAtIndex:](#) (page 15)

**Related Sample Code**

QTAudioContextInsert

QTAudioExtractionPanel

Quartz Composer WWDC 2005 TextEdit

UIElementInspector

WhackedTV

**Declared In**

NSPopupMenu.h

## menu

Returns the pop-up button's associated menu.

- (NSMenu \*)menu

### Return Value

The menu for the pop-up button.

### Availability

Available in Mac OS X v10.0 and later.

### Related Sample Code

ButtonMadness

OutputBins2PDE

QTAudioContextInsert

QTAudioExtractionPanel

Quartz Composer WWDC 2005 TextEdit

### Declared In

NSPopUpButton.h

## numberOfItems

Returns the number of items in the menu.

- (NSInteger)numberOfItems

### Return Value

The number of items in the menu.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- [lastItem](#) (page 17)

### Related Sample Code

MovieAssembler

QTAudioContextInsert

### Declared In

NSPopUpButton.h

## objectValue

Returns the index of the selected item.

- (id)objectValue

### Return Value

An object (typically an `NSNumber` object) that responds to the `intValue` message and contains the index of the selected item.

**See Also**

- [setObjectValue:](#) (page 25)

## preferredEdge

Returns the edge of the receiver next to which the pop-up menu is displayed under restrictive screen conditions.

- (NSRectEdge)preferredEdge

**Return Value**

Possible values include `NSMinXEdge`, `NSMinYEdge`, `NSMaxXEdge`, or `NSMaxYEdge`. The default value is the bottom edge, which is `NSMaxYEdge` for flipped views or `NSMinYEdge` for unflipped views.

**Discussion**

For pull-down menus, the default behavior is to display the menu under the receiver. For most pop-up menus, the `NSPopUpButton` object attempts to show the selected item directly over the button.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [setPreferredEdge:](#) (page 25)

**Declared In**

`NSPopUpButton.h`

## pullsDown

Returns a Boolean value indicating the behavior of the control's menu.

- (BOOL)pullsDown

**Return Value**

YES if the menu behaves like a pull-down menu; otherwise, NO if it behaves like a pop-up menu.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [setPullsDown:](#) (page 25)

**Declared In**

`NSPopUpButton.h`

## removeAllItems

Removes all items in the receiver's item menu.

- (void)removeAllItems

**Discussion**

After removing the items, this method uses the [synchronizeTitleAndSelectedItem](#) (page 26) method to refresh the menu.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [numberOfItems](#) (page 18)
- [removeItemAtIndex:](#) (page 20)
- [removeItemWithTitle:](#) (page 21)

**Related Sample Code**

QTAudioContextInsert  
 QTAudioExtractionPanel  
 Quartz Composer WWDC 2005 TextEdit  
 UIElementInspector  
 WhackedTV

**Declared In**

NSPopUpButton.h

**removeItemAtIndex:**

Removes the item at the specified index.

```
- (void)removeItemAtIndex:(NSInteger) index
```

**Parameters**

*index*

The zero-based index indicating which item to remove. Specifying 0 removes the item at the top of the menu.

**Discussion**

After removing the item, this method uses the [synchronizeTitleAndSelectedItem](#) (page 26) method to make sure the title displayed matches the currently selected item.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [insertItemWithTitle:atIndex:](#) (page 14)
- [removeAllItems](#) (page 19)
- [removeItemWithTitle:](#) (page 21)

**Related Sample Code**

CocoaSpeechSynthesisExample  
 QTAudioContextInsert

**Declared In**

NSPopUpButton.h

## removeItemWithTitle:

Removes the item with the specified title from the menu.

```
- (void)removeItemWithTitle:(NSString *)title
```

### Parameters

*title*

The title of the item you want to remove. If no menu item exists with the specified title, this method triggers an assertion.

### Discussion

This method removes the first item it finds with the specified name. This method then uses [synchronizeTitleAndSelectedItem](#) (page 26) to refresh the menu.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- [addItemWithTitle:](#) (page 9)
- [removeAllItems](#) (page 19)
- [removeItemAtIndex:](#) (page 20)

### Declared In

NSPopUpButton.h

## selectedItem

Returns the menu item last selected by the user.

```
- (NSMenuItem *)selectedItem
```

### Return Value

The menu item that is currently selected, or `nil` if no item is selected.

### Discussion

The last selected menu item is the one that was highlighted when the user released the mouse button. It is possible for a pull-down menu's selected item to be its first item.

### Availability

Available in Mac OS X v10.0 and later.

### Related Sample Code

EnhancedAudioBurn  
 FinalCutPro\_AppleEvents  
 QTAudioContextInsert  
 QTAudioExtractionPanel  
 Quartz Composer WWDC 2005 TextEdit

### Declared In

NSPopUpButton.h

**selectItem:**

Selects the specified menu item.

```
- (void)selectItem:(NSMenuItem *)item
```

**Parameters**

*anObject*

The menu item to select, or `nil` if you want to deselect all menu items.

**Availability**

Available in Mac OS X v10.0 and later.

**Related Sample Code**

NewsReader

QTAudioContextInsert

QTAudioExtractionPanel

WhackedTV

**Declared In**

NSPopUpButton.h

**selectItemAtIndex:**

Selects the item in the menu at the specified index.

```
- (void)selectItemAtIndex:(NSInteger)index
```

**Parameters**

*index*

The index of the item you want to select, or `-1` you want to deselect all menu items.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [indexOfSelectedItem](#) (page 13)

**Related Sample Code**

ColorMatching

GLUT

QTAudioContextInsert

Quartz Composer WWDC 2005 TextEdit

WhackedTV

**Declared In**

NSPopUpButton.h

**selectItemWithTag:**

Selects the menu item with the specified tag.

```
- (BOOL)selectItemWithTag:(NSInteger>tag
```

**Parameters***tag*

The tag of the item you want to select.

**Return Value**

YES if the item was successfully selected; otherwise, NO.

**Discussion**

If no item with the specified tag is found, this method returns NO and leaves the menu state unchanged.

You typically assign tags to menu items from Interface Builder, but you can also assign them programmatically using the `setTag:` method of `NSMenuItem`.

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

- [indexOfItemWithTag:](#) (page 11)

**Related Sample Code**

ScannerBrowser

WhackedTV

**Declared In**

NSPopUpButton.h

**selectItemWithTitle:**

Selects the item with the specified title.

```
- (void)selectItemWithTitle:(NSString *)title
```

**Parameters***title*

The title of the item to select. If you specify `nil`, an empty string, or a string that does not match the title of a menu item, this method deselects the currently selected item.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [indexOfItemWithTitle:](#) (page 12)

- [addItemWithTitle:](#) (page 9)

- [setTitle:](#) (page 26)

**Related Sample Code**

GLUT

**Declared In**

NSPopUpButton.h

## setAutoenablesItems:

Sets whether the receiver automatically enables and disables its items every time a user event occurs.

```
- (void)setAutoenablesItems:(BOOL)flag
```

### Parameters

*flag*

YES if you want the receiver to automatically enable and disable items; otherwise, NO.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- [autoenablesItems](#) (page 10)

### Related Sample Code

QTAudioContextInsert

QTAudioExtractionPanel

WhackedTV

### Declared In

NSPopUpButton.h

## setImage:

This method has no effect.

```
- (void)setImage:(NSImage *)anImage
```

### Parameters

*anImage*

The image to display.

### Discussion

The image displayed in a pop up button cell is taken from the selected menu item (in the case of a pop up menu) or from the first menu item (in the case of a pull-down menu).

## setMenu:

Sets the pop-up button's associated menu .

```
- (void)setMenu:(NSMenu *)menu
```

### Parameters

*menu*

The menu to associate with the pop-up button.

### Discussion

If another menu was already associated with the pop-up button, this method releases the old menu. If you want to explicitly save the old menu, you should retain it before invoking this method.

### Availability

Available in Mac OS X v10.0 and later.

**Related Sample Code**

ScannerBrowser

**Declared In**

NSPopUpButton.h

**setObjectValue:**

Selects the item at a specific index using an object value.

```
- (void)setObjectValue:(id)object
```

**Parameters***object*

An `NSNumber` object containing the index (an integer) of the item you want to select. Specify the index -1 to deselect all items. You can also use an object other than an `NSNumber` object. In that case, the object must respond to the `intValue` method and return an appropriate index value.

**Discussion****See Also**

- [objectValue](#) (page 18)

**setPreferredEdge:**

Sets the edge of the receiver next to which the pop-up menu should appear under restrictive screen conditions.

```
- (void)setPreferredEdge:(NSRectEdge)edge
```

**Parameters***edge*

The preferred edge. Possible values include `NSMinXEdge`, `NSMinYEdge`, `NSMaxXEdge`, or `NSMaxYEdge`.

**Discussion**

For pull-down menus, the default behavior is to display the menu under the receiver. For most pop-up menus, the `NSPopUpButton` object attempts to show the selected item directly over the button.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [preferredEdge](#) (page 19)

**Declared In**

NSPopUpButton.h

**setPullsDown:**

Sets whether the receiver behaves as a pull-down or pop-up menu.

```
- (void)setPullsDown:(BOOL)flag
```

**Parameters***flag*

YES if you want the receiver to operate as a pull-down menu; otherwise, NO if you want it to operate as a pop-up menu.

**Discussion**

This method does not change the contents of the menu; it changes only the style of the menu.

When changing the menu type to a pull-down menu, if the menu was a pop-up menu and the cell alters the state of its selected items, this method sets the state of the currently selected item to `NSStateOff` before changing the menu type.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [initWithFrame:pullsDown:](#) (page 13)
- [pullsDown](#) (page 19)

**Declared In**

NSPopUpButton.h

**setTitle:**

Sets the string displayed in the receiver when the user isn't pressing the mouse button.

```
- (void)setTitle:(NSString *)aString
```

**Parameters***aString*

The string to display.

**Discussion**

If the receiver displays a pop-up menu, this method changes the current item to be the item with the specified title, adding a new item by that name if one does not already exist. If the receiver displays a pull-down list, this method sets its title to the specified string.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSPopUpButton.h

**synchronizeTitleAndSelectedItem**

Ensures that the item being displayed by the receiver agrees with the selected item.

```
- (void)synchronizeTitleAndSelectedItem
```

**Discussion**

If there's no selected item, this method selects the first item in the item menu and sets the receiver's item to match. For pull-down menus, this method makes sure that the first item is being displayed (the `NSPopUpButtonCell` object must be set to use the selected menu item, which happens by default).

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [itemArray](#) (page 15)
- [indexOfSelectedItem](#) (page 13)

**Declared In**

NSPopUpButton.h

**titleOfSelectedItem**

Returns the title of the item last selected by the user.

```
- (NSString *)titleOfSelectedItem
```

**Return Value**

The title of the selected menu item, or an empty string if no item is selected.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [indexOfSelectedItem](#) (page 13)

**Related Sample Code**

AlbumToSlideshow  
FinalCutPro\_AppleEvents  
GLUT  
MovieAssembler  
WhackedTV

**Declared In**

NSPopUpButton.h

## Notifications

**NSPopUpButtonWillPopUpNotification**

Posted when an `NSPopUpButton` object receives a mouse-down event—that is, when the user is about to select an item from the menu.

The notification object is the selected `NSPopUpButton` object. This notification does not contain a `userInfo` dictionary.

**Availability**

Available in Mac OS X v10.0 and later.

**Declared In**

NSPopUpButton.h



# Document Revision History

---

This table describes the changes to *NSPopUpButton Class Reference*.

Date	Notes
2008-10-15	Replace parameter types of "id < MenuItem>" with "MenuItem *".
2006-05-23	First publication of this content as a separate document.

## REVISION HISTORY

### Document Revision History

# Index

---

## A

---

addItemWithTitles: [instance method 9](#)  
addItemWithTitle: [instance method 9](#)  
autoenablesItems [instance method 10](#)

## I

---

indexOfItem: [instance method 11](#)  
indexOfItemWithRepresentedObject: [instance method 11](#)  
indexOfItemWithTag: [instance method 11](#)  
indexOfItemWithTarget:andAction: [instance method 12](#)  
indexOfItemWithTitle: [instance method 12](#)  
indexOfSelectedItem [instance method 13](#)  
initWithFrame:pullsDown: [instance method 13](#)  
insertItemWithTitle:atIndex: [instance method 14](#)  
itemArray [instance method 15](#)  
itemAtIndex: [instance method 15](#)  
itemTitleAtIndex: [instance method 16](#)  
itemTitles [instance method 16](#)  
itemWithTitle: [instance method 17](#)

## L

---

lastItem [instance method 17](#)

## M

---

menu [instance method 18](#)

## N

---

NSPopUpButtonWillPopUpNotification [notification 27](#)  
numberOfItems [instance method 18](#)

## O

---

objectValue [instance method 18](#)

## P

---

preferredEdge [instance method 19](#)  
pullsDown [instance method 19](#)

## R

---

removeAllItems [instance method 19](#)  
removeItemAtIndex: [instance method 20](#)  
removeItemWithTitle: [instance method 21](#)

## S

---

selectedItem [instance method 21](#)  
selectItemAtIndex: [instance method 22](#)  
selectItem: [instance method 22](#)  
selectItemWithTag: [instance method 22](#)  
selectItemWithTitle: [instance method 23](#)  
setAutoenablesItems: [instance method 24](#)  
setImage: [instance method 24](#)  
setMenu: [instance method 24](#)  
setObjectValue: [instance method 25](#)  
setPreferredEdge: [instance method 25](#)  
setPullsDown: [instance method 25](#)  
setTitle: [instance method 26](#)

synchronizeTitleAndSelectedItem [instance method](#)  
[26](#)

## T

---

titleOfSelectedItem [instance method](#) [27](#)