
NSSliderCell Class Reference

[Cocoa](#) > [User Experience](#)



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

Shuffle is a trademark of Apple Inc.

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

NSSliderCell Class Reference 5

Overview	5
Tasks	5
Asking About the Cell's Behavior	5
Setting the Slider Type	6
Changing the Cell's Behavior	6
Displaying the Cell	6
Asking About the Cell's Appearance	6
Changing the Cell's Appearance	7
Asking About the Value Limits	7
Changing the Value Limits	7
Managing Tick Marks	7
Class Methods	8
prefersTrackingUntilMouseUp	8
Instance Methods	8
allowsTickMarkValuesOnly	8
altIncrementValue	9
closestTickMarkValueToValue:	9
drawBarInside:flipped:	10
drawKnob	10
drawKnob:	10
indexOfTickMarkAtPoint:	11
isVertical	11
knobRectFlipped:	12
knobThickness	12
maxValue	12
minValue	13
numberOfTickMarks	13
rectOfTickMarkAtIndex:	13
setAllowsTickMarkValuesOnly:	14
setAltIncrementValue:	14
setKnobThickness:	15
setMaxValue:	15
setMinValue:	16
setNumberOfTickMarks:	16
setSliderType:	17
setTickMarkPosition:	17
setTitle:	18
setTitleCell:	18
setTitleColor:	18
setTitleFont:	19

CONTENTS

sliderType	19
tickMarkPosition	19
tickMarkValueAtIndex:	20
title	20
titleCell	21
titleColor	21
titleFont	21
trackRect	21
Constants	22
NSTickMarkPosition	22
NSSliderType	23

Document Revision History 25

Index 27

NSSliderCell Class Reference

Inherits from	NSActionCell : NSCell : NSObject
Conforms to	NSCoding (NSCell) NSCopying (NSCell) NSObject (NSObject)
Framework	/System/Library/Frameworks/AppKit.framework
Availability	Available in Mac OS X v10.0 and later.
Companion guide	Slider Programming Topics for Cocoa
Declared in	NSSliderCell.h
Related sample code	AnimatedSlider QTKitMovieShuffler

Overview

An `NSSliderCell` object controls the appearance and behavior of an `NSSlider` object, or of a single slider in an `NSMatrix` object.

You can customize an `NSSliderCell` to a certain degree, using its `set...` methods. If these methods do not allow you sufficient flexibility, you can create a subclass. In that subclass, you can override any of the following methods: `knobRectFlipped:` (page 12), `drawBarInside:flipped:` (page 10), `drawKnob` (page 10), and `prefersTrackingUntilMouseUp` (page 8).

Tasks

Asking About the Cell's Behavior

- `altIncrementValue` (page 9)
Returns the amount by which the slider changes its value when the user drags with the Option key held down.
- + `prefersTrackingUntilMouseUp` (page 8)
Returns a Boolean value indicating whether the `NSSliderCell` continues to track the cursor until the next mouse up.

- [trackRect](#) (page 21)
Returns the rectangle within which the cell tracks the cursor while the mouse button is down.

Setting the Slider Type

- [setSliderType:](#) (page 17)
Sets the type of slider to a bar or a dial.
- [sliderType](#) (page 19)
Returns the slider type; either a bar or a dial.

Changing the Cell's Behavior

- [setAltIncrementValue:](#) (page 14)
Sets the amount by which the receiver modifies its value when the knob is Option-dragged.

Displaying the Cell

- [knobRectFlipped:](#) (page 12)
Returns the rectangle in which the slider knob is drawn.
- [drawBarInside:flipped:](#) (page 10)
Draws the slider's bar—but not its bezel or knob—inside the specified rectangle.
- [drawKnob](#) (page 10)
Calculates the rectangle in which the knob should be drawn, then invokes [drawKnob:](#) (page 10) to actually draw the knob.
- [drawKnob:](#) (page 10)
Draws the slider knob in the given rectangle.

Asking About the Cell's Appearance

- [knobThickness](#) (page 12)
Returns the slider knob's thickness, in pixels.
- [isVertical](#) (page 11)
Returns an integer indicating the orientation (vertical or horizontal) of the slider.
- [title](#) (page 20)
This method has been deprecated. Returns the slider's title.
- [titleCell](#) (page 21)
This method has been deprecated. Returns `nil`.
- [titleFont](#) (page 21)
This method has been deprecated. Returns `nil`.
- [titleColor](#) (page 21)
This method has been deprecated. Returns `nil`.

Changing the Cell's Appearance

- [setKnobThickness:](#) (page 15)
This method has been deprecated. Lets you set the knob's thickness, measured in pixels.
- [setTitle:](#) (page 18)
This method has been deprecated. Sets the title in the bar behind the slider's knob.
- [setTitleCell:](#) (page 18)
This method has been deprecated. Sets the cell used to draw the slider's title.
- [setTitleColor:](#) (page 18)
This method has been deprecated. Sets the color used to draw the slider's title.
- [setTitleFont:](#) (page 19)
This method has been deprecated. Sets the font used to draw the slider's title.

Asking About the Value Limits

- [maxValue](#) (page 12)
Returns the maximum value the slider can send to its target.
- [minValue](#) (page 13)
Returns the minimum value the slider can send to its target.

Changing the Value Limits

- [setMaxValue:](#) (page 15)
Sets the maximum value the slider can send to its target.
- [setMinValue:](#) (page 16)
Sets the minimum value the slider can send to its target.

Managing Tick Marks

- [allowsTickMarkValuesOnly](#) (page 8)
Returns a Boolean value indicating whether the receiver fixes its values to those values represented by its tick marks.
- [closestTickMarkValueToValue:](#) (page 9)
Returns the value of the tick mark closest to the specified value.
- [indexOfTickMarkAtPoint:](#) (page 11)
Returns the index of the tick mark closest to the location of the slider represented by the specified point.
- [numberOfTickMarks](#) (page 13)
Returns the number of tick marks associated with the slider.
- [rectOfTickMarkAtIndex:](#) (page 13)
Returns the bounding rectangle of the tick mark at the specified index.
- [setAllowsTickMarkValuesOnly:](#) (page 14)
Sets whether the receiver's values are fixed to the values represented by the tick marks.

- [setNumberOfTickMarks:](#) (page 16)
Sets the number of tick marks displayed by the receiver.
- [setTickMarkPosition:](#) (page 17)
Sets where tick marks appear relative to the receiver.
- [tickMarkPosition](#) (page 19)
Returns the position of the tick marks relative to the receiver.
- [tickMarkValueAtIndex:](#) (page 20)
Returns the receiver's value represented by the tick mark at the specified index.

Class Methods

prefersTrackingUntilMouseUp

Returns a Boolean value indicating whether the `NSSliderCell` continues to track the cursor until the next mouse up.

+ (BOOL)prefersTrackingUntilMouseUp

Return Value

YES if the `NSSliderCell` continues to track the cursor even after the cursor leaves the cell's tracking rectangle; otherwise, NO. By default, this method returns YES.

Discussion

If this method returns YES, this means that, once you take hold of a slider's knob (by putting the cursor inside the cell's frame rectangle and pressing the mouse button), you retain control of the knob until you release the mouse button, even if you drag the cursor clear to the other side of the screen.

Never call this method explicitly. Override it if you create a subclass of `NSSliderCell` that you want to track the mouse differently.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSSliderCell.h`

Instance Methods

allowsTickMarkValuesOnly

Returns a Boolean value indicating whether the receiver fixes its values to those values represented by its tick marks.

- (BOOL)allowsTickMarkValuesOnly

Return Value

YES if the slider's values are limited to those values represented by tick marks; otherwise, NO.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setAllowsTickMarkValuesOnly:](#) (page 14)

Declared In

NSSliderCell.h

altIncrementValue

Returns the amount by which the slider changes its value when the user drags with the Option key held down.

- (double)altIncrementValue

Return Value

The amount by which the slider changes its value when the user drags the knob with the Option key held down. Unless you call [setAltIncrementValue:](#) (page 14), [altIncrementValue](#) (page 9) returns -1.0, and the slider behaves no differently with the Option key down than with it up.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setAltIncrementValue:](#) (page 14)

Declared In

NSSliderCell.h

closestTickMarkValueToValue:

Returns the value of the tick mark closest to the specified value.

- (double)closestTickMarkValueToValue:(double)aValue

Parameters

aValue

The value for which to obtain the closest tick mark.

Return Value

The value of the closest tick mark.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

NSSliderCell.h

drawBarInside:flipped:

Draws the slider’s bar—but not its bezel or knob—inside the specified rectangle.

```
- (void)drawBarInside:(NSRect)aRect flipped:(BOOL)flipped
```

Parameters

aRect

The bounds of the slider's bar, not of its interior rectangle.

flipped

A Boolean value that indicates whether the cell’s control view—that is, the `NSSlider` or `NSMatrix` associated with the `NSSliderCell`—has a flipped coordinate system.

Discussion

You should never invoke this method explicitly. It’s included so you can override it in a subclass.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [drawKnob:](#) (page 10)

Declared In

`NSSliderCell.h`

drawKnob

Calculates the rectangle in which the knob should be drawn, then invokes [drawKnob:](#) (page 10) to actually draw the knob.

```
- (void)drawKnob
```

Discussion

Before this message is sent, a `lockFocus` method must be sent to the cell’s control view.

You might invoke this method if you override one of the display methods belonging to `NSControl` or `NSCell`.

If you create a subclass of `NSSliderCell`, don’t override this method. Override [drawKnob:](#) (page 10) instead.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSSliderCell.h`

drawKnob:

Draws the slider knob in the given rectangle.

```
- (void)drawKnob:(NSRect)knobRect
```

Parameters*knobRect*

The rectangle in which to draw the slider knob.

Discussion

Before this message is sent, a `lockFocus` message must be sent to the cell's control view.

You should never invoke this method explicitly. It's included so you can override it in a subclass.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSSliderCell.h`

indexOfTickMarkAtPoint:

Returns the index of the tick mark closest to the location of the slider represented by the specified point.

```
- (NSInteger)indexOfTickMarkAtPoint:(NSPoint)point
```

Parameters*point*

The point representing the slider location.

Return Value

The index of the tick mark closest to the specified location.

Discussion

If *point* is not within the bounding rectangle (plus an extra pixel of space) of any tick mark, the method returns `NSNotFound`. This method invokes [rectOfTickMarkAtIndex:](#) (page 13) for each tick mark on the slider until it finds a tick mark containing *point*.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSSliderCell.h`

isVertical

Returns an integer indicating the orientation (vertical or horizontal) of the slider.

```
- (NSInteger)isVertical
```

Return Value

1 if the slider is vertical, 0 if it's horizontal, and -1 if the orientation can't be determined (for example, if the slider hasn't been displayed yet). A slider is defined as vertical if its height is greater than its width.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSSliderCell.h`

knobRectFlipped:

Returns the rectangle in which the slider knob is drawn.

- (NSRect)knobRectFlipped:(BOOL)*flipped*

Parameters

flipped

YES if the coordinate system of the associated NSSlider or NSMatrix is flipped; otherwise NO. You can determine whether this is the case by sending the NSView message `isFlipped` message to the NSMatrix or NSSlider.

Return Value

The rectangle in which the knob is drawn, specified in the coordinate system of the NSSlider or NSMatrix with which the receiver is associated.

The knob rectangle depends on where in the slider the knob belongs—that is, it depends on the receiver’s minimum and maximum values and on the value the position of the knob will represent.

Discussion

You should never invoke this method explicitly. It’s included so you can override it in a subclass.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSSliderCell.h

knobThickness

Returns the slider knob’s thickness, in pixels.

- (CGFloat)knobThickness

Return Value

The thickness of the slider knob. The thickness is defined to be the extent of the knob along the long dimension of the bar. In a vertical slider, then, a knob’s thickness is its height; in a horizontal slider, its thickness is its width.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

NSSliderCell.h

maxValue

Returns the maximum value the slider can send to its target.

- (double)maxValue

Return Value

The maximum value of the slider. A horizontal slider sends its maximum value when the knob is at the right end of the slider; a vertical slider sends it when the knob is at the top. The maximum selectable value for a circular slider is just below `maxValue`; for example, if `maxValue` is 360, you can set the dial up to 359.999.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

`NSSliderCell.h`

minValue

Returns the minimum value the slider can send to its target.

- (double)minValue

Return Value

The minimum value of the slider. A vertical slider sends this value when its knob is at the bottom; a horizontal slider sends it when its knob is all the way to the left; a circular slider sends it when its knob is at the top.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSSliderCell.h`

numberOfTickMarks

Returns the number of tick marks associated with the slider.

- (NSInteger)numberOfTickMarks

Return Value

The number of tick marks. The tick marks assigned to the minimum and maximum values are included.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

`NSSliderCell.h`

rectOfTickMarkAtIndex:

Returns the bounding rectangle of the tick mark at the specified index.

- (NSRect)rectOfTickMarkAtIndex:(NSInteger) *index*

Parameters

index

The index of the tick mark for which to return the bounding rectangle. The minimum-value tick mark is at index 0.

Return Value

The bounding rectangle of the specified tick mark.

Discussion

If no tick mark is associated with *index*, the method raises `NSRangeException`.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

`NSSliderCell.h`

setAllowsTickMarkValuesOnly:

Sets whether the receiver's values are fixed to the values represented by the tick marks.

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

Parameters

flag

YES if the slider's values are fixed to the values represented by the slider's tick marks; otherwise NO. For example, if you specify YES for a slider that has a minimum value of 0, a maximum value of 100, and five markers, the allowable values are 0, 25, 50, 75, and 100. When users move the slider's knob, it jumps to the tick mark nearest the cursor when the mouse button is released. This method has no effect if the slider has no tick marks.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [allowsTickMarkValuesOnly](#) (page 8)

Declared In

`NSSliderCell.h`

setAltIncrementValue:

Sets the amount by which the receiver modifies its value when the knob is Option-dragged.

- (void)setAltIncrementValue:(double) *increment*

Parameters*increment*

The amount by which the receiver changes its value when the knob is Option-dragged. This number should be in the range of values the slider can represent—for example, if the slider has a minimum value of 5 and a maximum value of 10, *increment* should be between 0 and 5.

Discussion

If you don't call this method, the slider behaves the same with the Option key down as with it up. This is also the result when you call [setAltIncrementValue:](#) (page 14) with an increment of -1.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [maxValue](#) (page 12)
- [minValue](#) (page 13)

Declared In

NSSliderCell.h

setKnobThickness:

This method has been deprecated. Lets you set the knob's thickness, measured in pixels.

```
- (void)setKnobThickness:(CGFloat)thickness
```

Parameters*thickness*

The knob's thickness. The thickness is defined to be the extent of the knob along the long dimension of the bar. In a vertical slider, then, a knob's thickness is its height; in a horizontal slider, its thickness is its width.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [knobThickness](#) (page 12)

Declared In

NSSliderCell.h

setMaxValue:

Sets the maximum value the slider can send to its target.

```
- (void)setMaxValue:(double)aDouble
```

Parameters*aDouble*

The slider's maximum value. A horizontal slider sends its maximum value when its knob is all the way to the right; a vertical slider sends its maximum value when its knob is at the top. The maximum selectable value for a circular slider is just below `maxValue`; for example, if `maxValue` is 360, you can set the dial up to 359.999.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [maxValue](#) (page 12)

Declared In

NSSliderCell.h

setMinValue:

Sets the minimum value the slider can send to its target.

- (void)setMinValue:(double)aDouble

Parameters

aDouble

The slider's minimum value. A horizontal slider sends its minimum value when its knob is all the way to the left; a vertical slider sends its minimum value when its knob is at the bottom; a circular slider sends it when its knob is at the top.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [minValue](#) (page 13)

Declared In

NSSliderCell.h

setNumberOfTickMarks:

Sets the number of tick marks displayed by the receiver.

- (void)setNumberOfTickMarks:(NSInteger)numberOfTickMarks

Parameters

numberOfTickMarks

The number of tick marks displayed by the slider, including those assigned to the minimum and maximum values. By default, this value is 0, and no tick marks appear. The number of tick marks assigned to a slider, along with the slider's minimum and maximum values, determines the values associated with the tick marks.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [numberOfTickMarks](#) (page 13)

Related Sample Code

QTKitMovieShuffler

Declared In

NSSliderCell.h

setSliderType:

Sets the type of slider to a bar or a dial.

- (void)setSliderType:(NSSliderType)*sliderType***Parameters***sliderType*A constant indicating the type of the slider. Possible values are described in [NSTickMarkPosition](#) (page 22).**Discussion**

If *sliderType* is `NSCircularSlider`, then you get a fixed-size round slider. The minimum value (`minValue`) is at the top, and the value increases as you go clockwise around the dial. The maximum selectable value is just below `maxValue`; for example, if `maxValue` is 360, you can set the dial up to 359.999.

You can use the [setNumberOfTickMarks:](#) (page 16) method to display tick marks, and you can use the [setAllowsTickMarkValuesOnly:](#) (page 14) method to specify that values are limited to those values represented by tick marks. You can set this control to regular or small sizes; the mini size is not supported.

Availability

Available in Mac OS X v10.3 and later.

See Also

- [sliderType](#) (page 19)
- [setNumberOfTickMarks:](#) (page 16)
- [setAllowsTickMarkValuesOnly:](#) (page 14)

Related Sample Code

QTKitMovieShuffler

Declared In

NSSliderCell.h

setTickMarkPosition:

Sets where tick marks appear relative to the receiver.

- (void)setTickMarkPosition:(NSTickMarkPosition)*position***Parameters***position*A constant indicating the position of the tick marks. Possible values are described in [NSTickMarkPosition](#) (page 22).**Discussion**

This method has no effect if no tick marks have been assigned (that is, [numberOfTickMarks](#) (page 13) returns 0).

Availability

Available in Mac OS X v10.0 and later.

See Also

- [tickMarkPosition](#) (page 19)

Declared In

NSSliderCell.h

setTitle:

This method has been deprecated. Sets the title in the bar behind the slider's knob.

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

Parameters

title

The title.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [title](#) (page 20)

Declared In

NSSliderCell.h

setTitleCell:

This method has been deprecated. Sets the cell used to draw the slider's title.

```
- (void)setTitleCell:(NSCell *)aCell
```

Discussion

You only need to invoke this method if the default title cell, `NSTextFieldCell`, doesn't suit your needs—that is, if you want to display the title in a manner that `NSTextFieldCell` doesn't permit. When you do choose to override the default, *aCell* should be an instance of a subclass of `NSTextFieldCell`.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [titleCell](#) (page 21)

Declared In

NSSliderCell.h

setTitleColor:

This method has been deprecated. Sets the color used to draw the slider's title.

- (void)setTitleColor:(NSColor *)*color*

Availability

Available in Mac OS X v10.0 and later.

See Also

- [titleColor](#) (page 21)

Declared In

NSSliderCell.h

setTitleFont:

This method has been deprecated. Sets the font used to draw the slider's title.

- (void)setTitleFont:(NSFont *)*font*

Availability

Available in Mac OS X v10.0 and later.

See Also

- [titleFont](#) (page 21)

Declared In

NSSliderCell.h

sliderType

Returns the slider type; either a bar or a dial.

- (NSSliderType)sliderType

Return Value

A constant indicating the type of the slider. Possible return values are described in [NSSliderType](#) (page 23).

Availability

Available in Mac OS X v10.3 and later.

See Also

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

Declared In

NSSliderCell.h

tickMarkPosition

Returns the position of the tick marks relative to the receiver.

- (NSTickMarkPosition)tickMarkPosition

Return Value

A constant indicating the position of the tick marks. Possible values are described in [NSTickMarkPosition](#) (page 22). The default alignments are `NSTickMarkBelow` and `NSTickMarkLeft`.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

`NSSliderCell.h`

tickMarkValueAtIndex:

Returns the receiver's value represented by the tick mark at the specified index.

```
- (double)tickMarkValueAtIndex:(NSInteger) index
```

Parameters

index

The index of the tick mark for which to retrieve the value. The minimum-value tick mark has an index of 0.

Return Value

The value represented by the specified tick mark.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSSliderCell.h`

title

This method has been deprecated. Returns the slider's title.

```
- (NSString *)title
```

Return Value

The title. The default title is the empty string (@"").

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setTitle:](#) (page 18)

Declared In

`NSSliderCell.h`

titleCell

This method has been deprecated. Returns `nil`.

- (id)titleCell

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setTitleCell:](#) (page 18)

Declared In

NSSliderCell.h

titleColor

This method has been deprecated. Returns `nil`.

- (NSColor *)titleColor

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setTitleColor:](#) (page 18)

Declared In

NSSliderCell.h

titleFont

This method has been deprecated. Returns `nil`.

- (NSFont *)titleFont

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setTitleFont:](#) (page 19)

Declared In

NSSliderCell.h

trackRect

Returns the rectangle within which the cell tracks the cursor while the mouse button is down.

- (NSRect)trackRect

Return Value

The tracking rectangle of the `NSSliderCell`. This rectangle includes the slider bar, but not the bezel.

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSSliderCell.h`

Constants

NSTickMarkPosition

Specify where the tick marks of an `NSSliderCell` object appear.

```
typedef enum _NSTickMarkPosition {
    NSTickMarkBelow = 0,
    NSTickMarkAbove = 1,
    NSTickMarkLeft = NSTickMarkAbove,
    NSTickMarkRight = NSTickMarkBelow
} NSTickMarkPosition;
```

Constants

`NSTickMarkBelow`

Tick marks below (for horizontal sliders); the default for horizontal sliders.

Available in Mac OS X v10.0 and later.

Declared in `NSSliderCell.h`.

`NSTickMarkAbove`

Tick marks above (for horizontal sliders).

Available in Mac OS X v10.0 and later.

Declared in `NSSliderCell.h`.

`NSTickMarkLeft`

Tick marks to the left (for vertical sliders); the default for vertical sliders

Available in Mac OS X v10.0 and later.

Declared in `NSSliderCell.h`.

`NSTickMarkRight`

Tick marks to the right (for vertical sliders).

Available in Mac OS X v10.0 and later.

Declared in `NSSliderCell.h`.

Discussion

These constants are used in [setTickMarkPosition:](#) (page 17) and [tickMarkPosition](#) (page 19).

Availability

Available in Mac OS X v10.0 and later.

Declared In

`NSSliderCell.h`

NSSliderType

Define the types of sliders, used by [setSliderType:](#) (page 17) and [sliderType](#) (page 19).

```
typedef enum {  
    NSLinearSlider    = 0,  
    NSCircularSlider  = 1  
} NSSliderType;
```

Constants

NSLinearSlider

A bar-shaped slider.

Available in Mac OS X v10.3 and later.

Declared in NSSliderCell.h.

NSCircularSlider

A circular slider; that is, a dial.

Available in Mac OS X v10.3 and later.

Declared in NSSliderCell.h.

Availability

Available in Mac OS X v10.3 and later.

Declared In

NSSliderCell.h

Document Revision History

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

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

REVISION HISTORY

Document Revision History

Index

A

allowsTickMarkValuesOnly **instance method** [8](#)
altIncrementValue **instance method** [9](#)

C

closestTickMarkValueToValue: **instance method** [9](#)

D

drawBarInside:flipped: **instance method** [10](#)
drawKnob **instance method** [10](#)
drawKnob: **instance method** [10](#)

I

indexOfTickMarkAtPoint: **instance method** [11](#)
isVertical **instance method** [11](#)

K

knobRectFlipped: **instance method** [12](#)
knobThickness **instance method** [12](#)

M

maxValue **instance method** [12](#)
minValue **instance method** [13](#)

N

NSCircularSlider **constant** [23](#)
NSLinearSlider **constant** [23](#)
NSSliderType **data type** [23](#)
NSTickMarkAbove **constant** [22](#)
NSTickMarkBelow **constant** [22](#)
NSTickMarkLeft **constant** [22](#)
NSTickMarkPosition **data type** [22](#)
NSTickMarkRight **constant** [22](#)
numberOfTickMarks **instance method** [13](#)

P

prefersTrackingUntilMouseUp **class method** [8](#)

R

rectOfTickMarkAtIndex: **instance method** [13](#)

S

setAllowsTickMarkValuesOnly: **instance method** [14](#)
setAltIncrementValue: **instance method** [14](#)
setKnobThickness: **instance method** [15](#)
setMaxValue: **instance method** [15](#)
setMinValue: **instance method** [16](#)
setNumberOfTickMarks: **instance method** [16](#)
setSliderType: **instance method** [17](#)
setTickMarkPosition: **instance method** [17](#)
setTitleCell: **instance method** [18](#)
setTitle: **instance method** [18](#)
setTitleColor: **instance method** [18](#)
setTitleFont: **instance method** [19](#)
sliderType **instance method** [19](#)

T

tickMarkPosition instance method [19](#)
tickMarkValueAtIndex: instance method [20](#)
title instance method [20](#)
titleCell instance method [21](#)
titleColor instance method [21](#)
titleFont instance method [21](#)
trackRect instance method [21](#)