

---

# NSSlider 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.

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

## **NSSlider Class Reference 5**

---

Overview	5
Tasks	5
Asking About the Slider's Appearance	5
Changing the Slider's Appearance	6
Asking About the Slider's Title	6
Changing the Slider's Title	6
Asking About the Value Limits	6
Changing the Value Limits	7
Handling Mouse-down Events	7
Managing Tick Marks	7
Instance Methods	8
acceptsFirstMouse:	8
allowsTickMarkValuesOnly	8
altIncrementValue	8
closestTickMarkValueToValue:	9
image	9
indexOfTickMarkAtPoint:	10
isVertical	10
knobThickness	10
maxValue	11
minValue	11
numberOfTickMarks	12
rectOfTickMarkAtIndex:	12
setAllowsTickMarkValuesOnly:	12
setAltIncrementValue:	13
setImage:	13
setKnobThickness:	14
setMaxValue:	14
setMinValue:	15
setNumberOfTickMarks:	15
setTickMarkPosition:	16
setTitle:	16
setTitleCell:	16
setTitleColor:	17
setTitleFont:	17
tickMarkPosition	18
tickMarkValueAtIndex:	18
title	19
titleCell	19
titleColor	19

titleFont 19

**Document Revision History 21**

---

**Index 23**

---

# NSSlider Class Reference

---

<b>Inherits from</b>	NSControl : NSView : NSResponder : NSObject
<b>Conforms to</b>	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>	Slider Programming Topics for Cocoa
<b>Declared in</b>	NSSlider.h
<b>Related sample code</b>	OpenGLCompositorLab QTAudioExtractionPanel Sketch-112 URL CacheInfo WhackedTV

## Overview

An `NSSlider` object displays a range of values for something in the application. Sliders can be vertical or horizontal bars or circular dials. An indicator, or knob, notes the current setting. The user can move the knob in the slider's bar—or rotate the knob in a circular slider—to change the setting.

The `NSSlider` class uses the `NSSliderCell` class to implement its user interface.

## Tasks

### Asking About the Slider's Appearance

- [altIncrementValue](#) (page 8)  
Returns the amount by which the receiver changes its value when the user Option-drag the slider knob.
- [image](#) (page 9)  
This method has been deprecated. Returns `nil`.

- [knobThickness](#) (page 10)  
Returns the knob's thickness, in pixels.
- [isVertical](#) (page 10)  
Returns an integer indicating the orientation (horizontal or vertical) of the slider.

## Changing the Slider's Appearance

- [setAltIncrementValue:](#) (page 13)  
Sets the amount by which the `NSSliderCell` modifies its value when the user Option-drags the knob.
- [setImage:](#) (page 13)  
This method has been deprecated. Sets the image the receiver displays in the bar behind its knob.
- [setKnobThickness:](#) (page 14)  
This method has been deprecated. Lets you set the knob's thickness, measured in pixels.

## Asking About the Slider's Title

- [title](#) (page 19)  
Returns the receiver's title.
- [titleCell](#) (page 19)  
This method has been deprecated. Returns `nil`.
- [titleColor](#) (page 19)  
This method has been deprecated. Returns `nil`.
- [titleFont](#) (page 19)  
This method has been deprecated. Returns `nil`.

## Changing the Slider's Title

- [setTitle:](#) (page 16)  
This method has been deprecated. Sets the title the receiver displays in the bar behind its knob.
- [setTitleCell:](#) (page 16)  
This method has been deprecated. Sets the cell used to draw the receiver's title.
- [setTitleColor:](#) (page 17)  
This method has been deprecated. Sets the color used to draw the receiver's title.
- [setTitleFont:](#) (page 17)  
This method has been deprecated. Sets the font used to draw the receiver's title.

## Asking About the Value Limits

- [maxValue](#) (page 11)  
Returns the maximum value the receiver can send to its target.
- [minValue](#) (page 11)  
Returns the minimum value the receiver can send to its target.

## Changing the Value Limits

- `setMaxValue:` (page 14)  
Sets the maximum value the receiver can send to its target.
- `setMinValue:` (page 15)  
Sets the minimum value the receiver can send to its target

## Handling Mouse-down Events

- `acceptsFirstMouse:` (page 8)  
Returns a Boolean value indicating whether the slider accepts a single mouse-down event that simultaneously activates the window and takes hold of the slider's knob.

## 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 10)  
Returns the index of the tick mark closest to the location of the receiver represented by the given point.
- `numberOfTickMarks` (page 12)  
Returns the number of tick marks associated with the receiver.
- `rectOfTickMarkAtIndex:` (page 12)  
Returns the bounding rectangle of the tick mark at the given index.
- `setAllowsTickMarkValuesOnly:` (page 12)  
Sets whether the receiver's values are fixed to the values represented by the tick marks.
- `setNumberOfTickMarks:` (page 15)  
Sets the number of tick marks displayed by the receiver.
- `setTickMarkPosition:` (page 16)  
Sets where tick marks appear relative to the receiver.
- `tickMarkPosition` (page 18)  
Returns how the receiver's tick marks are aligned with it.
- `tickMarkValueAtIndex:` (page 18)  
Returns the receiver's value represented by the tick mark at the specified index.

## Instance Methods

### acceptsFirstMouse:

Returns a Boolean value indicating whether the slider accepts a single mouse-down event that simultaneously activates the window and takes hold of the slider's knob.

- (BOOL)acceptsFirstMouse:(NSEvent \*)*mouseDownEvent*

#### Parameters

*mouseDownEvent*

The mouse-down event.

#### Return Value

YES if the receiver accepts the first mouse-down event; otherwise, NO. Returns YES by default.

#### Discussion

If you want the receiver to wait for its own mouse-down event, you must override this method.

#### Availability

Available in Mac OS X v10.0 and later.

#### Declared In

NSSlider.h

### 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 fixes its values to the values represented by its tick marks; otherwise, NO.

#### Discussion

In its implementation of this method, the receiving `NSSlider` object simply invokes the method of the same name of its `NSSliderCell` object.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

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

#### Declared In

NSSlider.h

### altIncrementValue

Returns the amount by which the receiver changes its value when the user Option-draws the slider knob.



- (double)altIncrementValue

#### Return Value

The amount by which the value changes when the user drags the slider knob with the Option key held down. Unless you call [setAltIncrementValue:](#) (page 13), [altIncrementValue](#) (page 8) returns -1.0, and the receiver 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 13)

#### Declared In

NSSlider.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 return the closest tick mark.

#### Return Value

The value of the tick mark closest to *aValue*.

#### Discussion

In its implementation of this method, the receiver simply invokes the method of the same name of its `NSSliderCell` instance.

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

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

#### Declared In

NSSlider.h

## image

This method has been deprecated. Returns `nil`.

- (NSImage \*)image

#### Availability

Available in Mac OS X v10.0 and later.

#### See Also

- [setImage:](#) (page 13)

**Declared In**  
NSSlider.h

## indexOfTickMarkAtPoint:

Returns the index of the tick mark closest to the location of the receiver represented by the given point.

- (NSInteger)indexOfTickMarkAtPoint:(NSPoint)point

### Parameters

*point*

The point representing the location for which to retrieve the tick mark.

### Return Value

The index of the tick mark closest to the location specified by *point*. If *point* is not within the bounding rectangle (plus an extra pixel of space) of any tick mark, the method returns `NSNotFound`.

### Discussion

In its implementation of this method, the receiving `NSSlider` instance simply invokes the method of the same name of its `NSSliderCell` instance. This method invokes [rectOfTickMarkAtIndex:](#) (page 12) for each tick mark on the slider until it finds a tick mark containing the point.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

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

**Declared In**  
NSSlider.h

## isVertical

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

- (NSInteger)isVertical

### Return Value

1 if the receiver 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**  
NSSlider.h

## knobThickness

Returns the 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, a knob's thickness is its width.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

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

### Declared In

NSSlider.h

## maxValue

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

- (double)maxVaLue

### Return Value

The slider's maximum value. A horizontal slider sends its maximum value when the knob is at the right end of the bar; a vertical slider sends it when the knob is at the top.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

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

### Declared In

NSSlider.h

## minValue

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

- (double)minVaLue

### Return Value

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

### Availability

Available in Mac OS X v10.0 and later.

### See Also

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

### Declared In

NSSlider.h

## numberOfTickMarks

Returns the number of tick marks associated with the receiver.

- (NSInteger)numberOfTickMarks

### Return Value

The number of the slider's tick marks. The tick marks assigned to the minimum and maximum values are included. In its implementation of this method, the receiving `NSSlider` instance simply invokes the method of the same name of its `NSSliderCell` instance.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

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

### Declared In

`NSSlider.h`

## rectOfTickMarkAtIndex:

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

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

### Parameters

*index*

The index of the tick mark for which to retrieve the bounds. 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`. In its implementation of this method, the receiving `NSSlider` instance simply invokes the method of the same name of its `NSSliderCell` instance.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

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

### Declared In

`NSSlider.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 should be fixed to the values represented by its tick marks; otherwise NO. For example, if a slider 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.

**Discussion**

This method has no effect if the slider has no tick marks. In its implementation of this method, the receiving `NSSlider` instance simply invokes the method of the same name of its `NSSliderCell` instance.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [allowsTickMarkValuesOnly](#) (page 8)

**Declared In**

`NSSlider.h`

**setAltIncrementValue:**

Sets the amount by which the `NSSliderCell` modifies its value when the user Option-drag the knob.

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

**Parameters***increment*

The amount by which the slider's value changes when the user Option-drag its knob. This value must fit 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. If *increment* is outside that range, the value is unchanged.

**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:` with an increment of `-1`.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [maxValue](#) (page 11)

- [minValue](#) (page 11)

**Declared In**

`NSSlider.h`

**setImage:**

This method has been deprecated. Sets the image the receiver displays in the bar behind its knob.

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

**Parameters***barImage*

The image to set.

**Discussion**The slider may scale and distort *barImage* to fit inside the bar.

The knob may cover part of the image. If you want the image to be visible all the time, you're better off placing it near the slider.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**- [setImage:](#) (page 13)**Declared In**

NSSlider.h

**setKnobThickness:**

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

- (void)setKnobThickness:(CGFloat)*thickness***Parameters***thickness*

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

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**- [knobThickness](#) (page 10)**Declared In**

NSSlider.h

**setMaxValue:**

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

- (void)setMaxVaLue:(double)*maxVaLue***Parameters***maxVaLue*

The maximum value of the slider. 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.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [maxValue](#) (page 11)

**Declared In**

NSSlider.h

**setMinValue:**

Sets the minimum value the receiver can send to its target

- (void)setMinValue:(double)minValue

**Parameters**

*minValue*

The minimum value of the slider. 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.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [minValue](#) (page 11)

**Declared In**

NSSlider.h

**setNumberOfTickMarks:**

Sets the number of tick marks displayed by the receiver.

- (void)setNumberOfTickMarks:(NSInteger)numberOfTickMarks

**Parameters**

*numberOfTickMarks*

The number of tick marks (including those assigned to the minimum and maximum values) displayed by the slider. 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.

**Discussion**

In its implementation of this method, the receiving `NSSlider` instance simply invokes the method of the same name of its `NSSliderCell` instance.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [numberOfTickMarks](#) (page 12)

**Declared In**

NSSlider.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. For horizontal sliders, this can be `NSTickMarkBelow` (the default) or `NSTickMarkAbove`; for vertical sliders, this can be `NSTickMarkLeft` (the default) or `NSTickMarkRight`.

### Discussion

This method has no effect if no tick marks have been assigned (that is, `numberOfTickMarks` (page 12) returns 0). In its implementation of this method, the receiving `NSSlider` instance simply invokes the method of the same name of its `NSSliderCell` instance.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- [tickMarkPosition](#) (page 18)

### Declared In

`NSSlider.h`

## setTitle:

This method has been deprecated. Sets the title the receiver displays in the bar behind its knob.

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

### Parameters

*barTitle*

The slider's title. The knob may cover part or all of the title. If you want the title to be visible all the time, you're better off placing a label near the slider.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

- [title](#) (page 19)

### Declared In

`NSSlider.h`

## setTitleCell:

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

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



**Parameters***titleCell*

The cell used to draw the title.

**Discussion**

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

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [titleCell](#) (page 19)

**Declared In**

`NSSlider.h`

**setTitleColor:**

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

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

**Parameters***color*

The title color.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [titleColor](#) (page 19)

**Declared In**

`NSSlider.h`

**setFont:**

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

```
- (void)setFont:(NSFont *)font
```

**Parameters***font*

The title font.

**Availability**

Available in Mac OS X v10.0 and later.

**See Also**

- [setFont](#) (page 19)

**Declared In**  
NSSlider.h

## tickMarkPosition

Returns how the receiver's tick marks are aligned with it.

- (NSTickMarkPosition)tickMarkPosition

### Return Value

A constant indicating the position of the tick marks. Possible values are NSTickMarkBelow, NSTickMarkAbove, NSTickMarkLeft, and NSTickMarkRight (the last two are for vertical sliders). The default alignments are NSTickMarkBelow and NSTickMarkLeft.

### Discussion

In its implementation of this method, the receiving NSSlider instance simply invokes the method of the same name of its NSSliderCell instance.

### Availability

Available in Mac OS X v10.0 and later.

### See Also

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

**Declared In**  
NSSlider.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 return the value. The minimum-value tick mark has an index of 0.

### Return Value

The value of the specified tick mark.

### Discussion

In its implementation of this method, the receiving NSSlider instance simply invokes the method of the same name of its NSSliderCell instance.

### Availability

Available in Mac OS X v10.0 and later.

**Declared In**  
NSSlider.h

## title

Returns the receiver'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 16)

### Declared In

NSSlider.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 16)

### Declared In

NSSlider.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 17)

### Declared In

NSSlider.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 17)

**Declared In**

NSSlider.h

# Document Revision History

---

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

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

## REVISION HISTORY

### Document Revision History

# Index

---

## A

---

acceptsFirstMouse: [instance method 8](#)  
allowsTickMarkValuesOnly [instance method 8](#)  
altIncrementValue [instance method 8](#)

## C

---

closestTickMarkValueToValue: [instance method 9](#)

## I

---

image [instance method 9](#)  
indexOfTickMarkAtPoint: [instance method 10](#)  
isVertical [instance method 10](#)

## K

---

knobThickness [instance method 10](#)

## M

---

maxValue [instance method 11](#)  
minValue [instance method 11](#)

## N

---

numberOfTickMarks [instance method 12](#)

## R

---

rectOfTickMarkAtIndex: [instance method 12](#)

## S

---

setAllowsTickMarkValuesOnly: [instance method 12](#)  
setAltIncrementValue: [instance method 13](#)  
setImage: [instance method 13](#)  
setKnobThickness: [instance method 14](#)  
setMaxValue: [instance method 14](#)  
setMinValue: [instance method 15](#)  
setNumberOfTickMarks: [instance method 15](#)  
setTickMarkPosition: [instance method 16](#)  
setTitleCell: [instance method 16](#)  
setTitle: [instance method 16](#)  
setTitleColor: [instance method 17](#)  
setTitleFont: [instance method 17](#)

## T

---

tickMarkPosition [instance method 18](#)  
tickMarkValueAtIndex: [instance method 18](#)  
title [instance method 19](#)  
titleCell [instance method 19](#)  
titleColor [instance method 19](#)  
titleFont [instance method 19](#)