

---

# NSFormCell

<b>Inherits From:</b>	NSActionCell : NSCell : NSObject
<b>Conforms To:</b>	NSCoding (from NSCell) NSCopying (from NSCell) NSObject (from NSObject)
<b>Declared In:</b>	AppKit/NSFormCell.h

## Class Description

This class is used to implement text entry fields in an NSForm. The left part of an NSFormCell is a title. The right part is an editable text entry field.

For more on the use of NSFormCell, see the class specification for NSForm.

## Method Types

Initializing an NSFormCell	– initTextCell:
Asking about a cell's appearance	– isOpaque
Asking about a cell's title	– attributedTitle – title – titleAlignment – titleFont – titleWidth
Changing the cell's title	– setAttributedTitle: – setTitle: – setTitleAlignment: – setTitleFont: – setTitleWidth:
Setting a keyboard equivalent	– setTitleWithMnemonic:

## Instance Methods



### **attributedTitle**

– (NSAttributedString \*)**attributedTitle**

Returns the title as an attributed string.

### **initWithCell:**

– (id)**initWithCell:**(NSString \*)*aString*

Initializes a newly allocated NSFormCell. Its title is set to *aString*; the contents of its text entry field are set to the empty string (“”). The font for both title and text is the user’s chosen system font in 12.0 point, and the text area is drawn with a bezel. This method is the designated initializer for NSFormCell.

Returns self.

**See also:** – **setTitle:**

### **isOpaque**

– (BOOL)**isOpaque**

Returns YES if both the title and the entry field are opaque, NO if one or both of them are transparent. Since titles are transparent by default, this method usually returns NO.



### **setAttributedTitle:**

– (void)**setAttributedTitle:**(NSAttributedString \*)*anAttributedString*

Sets the cell’s title and title attributes according to *anAttributedString*.

### **setTitle:**

– (void)**setTitle:**(NSString \*)*aString*

Sets the cell’s title to *aString*.

### **setTitleAlignment:**

– (void)**setTitleAlignment:**(NSTextAlignment)*alignment*

Sets the alignment of the title. *alignment* can be one of three constants: NSTextAlignmentLeft, NSTextAlignmentRight, or NSTextAlignmentCenter.

---

### **setTitleFont:**

– (void)**setTitleFont:**(NSFont \*)*font*

Sets the title’s font.

### **setTitleWidth:**

– (void)**setTitleWidth:**(float)*width*

You usually won’t need to invoke this method, since the Application Kit automatically sets the title width whenever the title changes. If, however, the automatic width doesn’t suit your needs, you can use **setTitleWidth:** to set the width in pixels.

Once you have set the width explicitly this way, the Application Kit stops setting the width automatically; you will need to invoke **setTitleWidth:** every time the title changes. If you want the Application Kit to resume automatic width assignments, invoke **setTitleWidth:** with a negative *width* value.



### **setTitleWithMnemonic:**

– (void)**setTitleWithMnemonic:**(NSString \*)*titleWithAmpersand*

Sets the cell title and a single mnemonic character. The mnemonic character, which follows the ampersand in *titleWithAmpersand*, serves as an Alt-key equivalent to clicking in the text entry field.

For example, if *titleWithAmpersand* is “T&title,” the cell’s title will be displayed as “Title” (the mnemonic character, *i*, is underlined). If a user types Alt-i, it will have the same effect as clicking in the text entry field.

**See also:** – **setTitle:**

### **title**

– (NSString \*)**title**

Returns the cell’s title. The default title is “Field:”.

### **titleAlignment**

– (NSTextAlignment)**titleAlignment**

Returns the alignment of the title, which will be one of the following: `NSLeftTextAlignment`, `NSCenterTextAlignment`, or `NSRightTextAlignment` (the default).

### **titleFont**

– (NSFont \*)**titleFont**

Returns the font used to draw the cell's title.

### **titleWidth**

– (float)**titleWidth**

Returns the width (in pixels) of the title field. If you specified the width using **setTitleWidth:**, **titleWidth** returns the value you chose. Otherwise, it returns the width calculated automatically by the Application Kit.