
NSForm Class Reference

User Experience: Data Presentation



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

NSForm Class Reference 5

Overview	5
Tasks	5
Adding and Removing Entries	5
Changing the Appearance of All the Entries	6
Getting Cells and Indices	6
Displaying a Cell	6
Editing Text	7
Instance Methods	7
addEntry:	7
cellAtIndex:	7
drawCellAtIndex:	8
indexOfCellWithTag:	8
indexOfSelectedItem	9
insertEntry:atIndex:	9
removeEntryAtIndex:	9
selectTextAtIndex:	10
setBezeled:	10
setBordered:	10
setEntryWidth:	11
setFrameSize:	11
setInterlineSpacing:	12
setTextAlignment:	12
setTextBaseWritingDirection:	12
setFont:	13
setTitleAlignment:	13
setTitleBaseWritingDirection:	13
setTitleFont:	14

Document Revision History 15

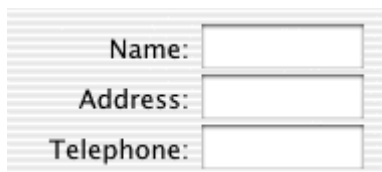
Index 17

NSForm Class Reference

Inherits from	NSMatrix : NSControl : NSView : NSResponder : NSObject
Conforms to	NSUserInterfaceValidations (NSMatrix) NSAnimatablePropertyContainer (NSView) NSCoding (NSResponder) NSObject (NSObject)
Framework	/System/Library/Frameworks/AppKit.framework
Availability	Available in Mac OS X v10.0 and later.
Declared in	NSForm.h
Companion guides	Forms Matrix Programming Guide for Cocoa
Related sample code	SimpleComboBox SourceView

Overview

An `NSForm` object is a vertical matrix of `NSFormCell` objects. Here's an example:



The image shows a vertical matrix of three rows. Each row consists of a text label followed by a rectangular input field. The labels are "Name:", "Address:", and "Telephone:" from top to bottom. The input fields are empty.

An `NSForm` object uses `NSFormCell` to implement its user interface.

Tasks

Adding and Removing Entries

- [addEntry:](#) (page 7)

Adds a new entry to the end of the receiver and gives it the specified title.

- [insertEntry:atIndex:](#) (page 9)
Inserts an entry with the specified title into the receiver.
- [removeEntryAtIndex:](#) (page 9)
Removes and releases the entry at the specified index.

Changing the Appearance of All the Entries

- [setBezeled:](#) (page 10)
Sets whether the receiver's entries should display a bezel around their editable text.
- [setBordered:](#) (page 10)
Sets whether the receiver's entries should display a border around their editable text fields.
- [setEntryWidth:](#) (page 11)
Sets the width of all the entries in the receiver.
- [setFrameSize:](#) (page 11)
Sets the size of the receiver's frame size to the specified value.
- [setInterlineSpacing:](#) (page 12)
Sets the spacing between entries
- [setTitleAlignment:](#) (page 13)
Sets the alignment for all of the entry titles.
- [setTitleBaseWritingDirection:](#) (page 13)
Sets the writing direction for the title of every control embedded in the form.
- [setTextAlignment:](#) (page 12)
Sets the alignment for all of the receiver's editable text.
- [setTextBaseWritingDirection:](#) (page 12)
Sets the writing direction for the text content of every control embedded in the form.
- [setTitleFont:](#) (page 14)
Sets the font for all of the entry titles.
- [setTextFont:](#) (page 13)
Sets the font for all of the receiver's editable text fields

Getting Cells and Indices

- [indexOfCellWithTag:](#) (page 8)
Returns the index of the entry whose tag is *tag*.
- [indexOfSelectedItem](#) (page 9)
Returns the index of the selected entry.
- [cellAtIndex:](#) (page 7)
Returns the entry at the specified index.

Displaying a Cell

- [drawCellAtIndex:](#) (page 8)
Displays the entry at the specified index.

Editing Text

- [selectTextAtIndex:](#) (page 10)
Selects the entry at the specified index.

Instance Methods

addEntry:

Adds a new entry to the end of the receiver and gives it the specified title.

```
(NSFormCell *)addEntry:(NSString *)title
```

Parameters

title

The title for the new form entry.

Return Value

The form cell object that was created for the entry.

Discussion

The new entry has no tag, target, or action, but is enabled and editable.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [insertEntryAtIndex:](#) (page 9)
- `setEditable:` (NSCell)
- `setTag:` (NSActionCell)
- `setTarget:` (NSActionCell)
- `setAction:` (NSActionCell)
- `setEnabled:` (NSActionCell)

Declared In

NSForm.h

cellAtIndex:

Returns the entry at the specified index.

```
(id)cellAtIndex:(NSInteger)entryIndex
```

Parameters

entryIndex

The index of the desired entry.

Return Value

The form cell object at the specified index.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [indexOfCellWithTag](#): (page 8)
- [indexOfSelectedItem](#) (page 9)

Declared In

NSForm.h

drawCellAtIndex:

Displays the entry at the specified index.

```
- (void)drawCellAtIndex:(NSInteger)entryIndex
```

Parameters

entryIndex

The index of the entry to draw.

Discussion

Because this method is called automatically whenever a cell needs drawing, you never need to invoke it explicitly. It is included in the API so you can override it if you subclass `NSFormCell`.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [indexOfCellWithTag](#): (page 8)
- [indexOfSelectedItem](#) (page 9)

Declared In

NSForm.h

indexOfCellWithTag:

Returns the index of the entry whose tag is *tag*.

```
- (NSInteger)indexOfCellWithTag:(NSInteger)tag
```

Parameters

tag

The tag of the desired entry.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [tag](#) (NSCell)

Declared In

NSForm.h

indexOfSelectedItem

Returns the index of the selected entry.

- (NSInteger)indexOfSelectedItem

Return Value

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

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSForm.h

insertEntryAtIndex:

Inserts an entry with the specified title into the receiver.

- (NSFormCell *)insertEntry:(NSString *)*title* atIndex:(NSInteger)*entryIndex*

Parameters

title

The title for the new form entry.

entryIndex

The zero-based index at which to insert the entry.

Return Value

The form cell object that was created for the entry.

Discussion

The new entry has no tag, target, or action, but is enabled and editable.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [addEntry:](#) (page 7)

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

Declared In

NSForm.h

removeEntryAtIndex:

Removes and releases the entry at the specified index.

- (void)removeEntryAtIndex:(NSInteger)*entryIndex*

Parameters

entryIndex

The zero-based index identifying the desired entry.

Discussion

If the specified index is invalid, this method does nothing.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSForm.h

selectTextAtIndex:

Selects the entry at the specified index.

```
- (void)selectTextAtIndex:(NSInteger)entryIndex
```

Parameters

entryIndex

The index of the entry to select. If the specified index is invalid, this method does nothing.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSForm.h

setBezeled:

Sets whether the receiver's entries should display a bezel around their editable text.

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

Parameters

flag

YES to display a bezel around all entries; otherwise, NO to show no bezel around all entries.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setBordered:](#) (page 10)
- [isBezeled](#) (NSCell)

Declared In

NSForm.h

setBordered:

Sets whether the receiver's entries should display a border around their editable text fields.

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

Parameters*flag*

YES to display a border around all entries; otherwise, NO to show no border around all entries.

Discussion

The border is drawn as a thin line around the editable text field. An entry can have a border or a bezel, but not both.

Availability

Available in Mac OS X v10.0 and later.

See Also

- [setBezeled:](#) (page 10)
- `isBordered` (NSCell)

Declared In

NSForm.h

setEntryWidth:

Sets the width of all the entries in the receiver.

```
- (void)setEntryWidth:(CGFloat)width
```

Parameters*width*

The width of all entries, measured in points in the user coordinate space. This value represents the width of both the title and the text field.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSForm.h

setFrameSize:

Sets the size of the receiver's frame size to the specified value.

```
- (void)setFrameSize:(NSSize)newSize
```

Parameters*newSize*

The new size of the form.

Discussion

The width of `NSFormCell` objects always match the width of their encompassing `NSForm` object. The cell width is always changed to match the view regardless of the value returned by `autosizesCells`.

Availability

Available in Mac OS X v10.3 and later.

Declared In

NSForm.h

setInterlineSpacing:

Sets the spacing between entries

- (void)setInterlineSpacing:(CGFloat)*spacing*

Parameters

spacing

The spacing between entries, measured in points in the user coordinate space.

Availability

Available in Mac OS X v10.0 and later.

Declared In

NSForm.h

setTextAlignment:

Sets the alignment for all of the receiver's editable text.

- (void)setTextAlignment:(NSInteger)*alignment*

Parameters

alignment

The alignment can be one of the following constants: NSRightTextAlignment, NSCenterTextAlignment, or NSLeftTextAlignment.

Discussion

The default alignment is NSLeftTextAlignment.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

NSForm.h

setTextBaseWritingDirection:

Sets the writing direction for the text content of every control embedded in the form.

- (void)setTextBaseWritingDirection:(NSWritingDirection)*writingDirection*

Parameters

writingDirection

This value can be one of the following constants: NSWritingDirectionNatural, NSWritingDirectionLeftToRight, or NSWritingDirectionRightToLeft.

Availability

Available in Mac OS X v10.4 and later.

See Also

- [baseWritingDirection](#) (NSControl)

Declared In

NSForm.h

setTextFont:

Sets the font for all of the receiver's editable text fields

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

Parameters

font

The font to use for all editable text fields.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

NSForm.h

setTitleAlignment:

Sets the alignment for all of the entry titles.

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

Parameters

alignment

The alignment can be one of the following constants: `NSRightTextAlignment`, `NSCenterTextAlignment`, or `NSLeftTextAlignment`.

Discussion

The default alignment is `NSRightTextAlignment`.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

NSForm.h

setTitleBaseWritingDirection:

Sets the writing direction for the title of every control embedded in the form.

- (void)setTitleBaseWritingDirection:(NSWritingDirection)*writingDirection*

Parameters

writingDirection

This value can be one of the following constants: `NSWritingDirectionNatural`, `NSWritingDirectionLeftToRight`, or `NSWritingDirectionRightToLeft`.

Availability

Available in Mac OS X v10.4 and later.

Declared In

NSForm.h

setTitleFont:

Sets the font for all of the entry titles.

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

Parameters

font

The font to use for all entry titles.

Availability

Available in Mac OS X v10.0 and later.

See Also

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

Declared In

NSForm.h

Document Revision History

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

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

REVISION HISTORY

Document Revision History

Index

A

addEntry: [instance method 7](#)

C

cellAtIndex: [instance method 7](#)

D

drawCellAtIndex: [instance method 8](#)

I

indexOfCellWithTag: [instance method 8](#)

indexOfSelectedItem: [instance method 9](#)

insertEntryAtIndex: [instance method 9](#)

R

removeEntryAtIndex: [instance method 9](#)

S

selectTextAtIndex: [instance method 10](#)

setBezeled: [instance method 10](#)

setBordered: [instance method 10](#)

setEntryWidth: [instance method 11](#)

setFrameSize: [instance method 11](#)

setInterlineSpacing: [instance method 12](#)

setTextAlignment: [instance method 12](#)

setTextBaseWritingDirection: [instance method 12](#)

setTextFont: [instance method 13](#)

setTitleAlignment: [instance method 13](#)

setTitleBaseWritingDirection: [instance method 13](#)

setTitleFont: [instance method 14](#)