

---

# NSTextFieldCell Class Reference

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



2007-03-05



Apple Inc.  
© 2007 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, 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

## **NSTextFieldCell Class Reference 5**

---

Overview 5

Tasks 5

Managing the Token Style 5

Managing the Tokenizing Character Set 5

Configuring the Completion Delay 6

Managing the Delegate 6

Displaying Tokenized Strings 6

Editing Tokenized Strings 6

Reading From and Writing to the Pasteboard 6

Managing Menus for Represented Objects 7

Class Methods 7

defaultCompletionDelay 7

defaultTokenizingCharacterSet 7

Instance Methods 8

completionDelay 8

delegate 8

setCompletionDelay: 8

setDelegate: 9

setTokenizingCharacterSet: 9

setTokenStyle: 9

tokenizingCharacterSet 10

tokenStyle 10

Delegate Methods 11

tokenFieldCell:completionsForSubstring:indexOfToken:indexOfSelectedItem: 11

tokenFieldCell:displayStringForRepresentedObject: 11

tokenFieldCell:editingStringForRepresentedObject: 12

tokenFieldCell:hasMenuForRepresentedObject: 12

tokenFieldCell:menuForRepresentedObject: 12

tokenFieldCell:readFromPasteboard: 12

tokenFieldCell:representedObjectForEditingString: 13

tokenFieldCell:shouldAddObjects:atIndex: 13

tokenFieldCell:styleForRepresentedObject: 14

tokenFieldCell:writeRepresentedObjects:toPasteboard: 14

Constants 14

NSTokenStyle 14

NSTokenStyle Values 15

**Document Revision History 17**

---

**Index 19**

---

# NSTextFieldCell Class Reference

---

<b>Inherits from</b>	NSTextFieldCell : NSActionCell : NSCell : NSObject
<b>Conforms to</b>	NSCoding (NSCell) NSCopying (NSCell) NSObject (NSObject)
<b>Framework</b>	/System/Library/Frameworks/AppKit.framework
<b>Availability</b>	Available in Mac OS X v10.4 and later.
<b>Companion guide</b>	Token Field Programming Guide for Cocoa
<b>Declared in</b>	NSTextFieldCell.h

## Overview

`NSTextFieldCell` is a subclass of `NSTextField` that provides tokenized editing of an array of objects similar to the address field in the Mail application. The objects may be strings or objects that can be represented as strings. A single token field cell can be presented in an `NSTextField` control.

## Tasks

### Managing the Token Style

- [setTokenStyle:](#) (page 9)  
Sets the token style of the receiver.
- [tokenStyle](#) (page 10)  
Returns the receiver's token style.

### Managing the Tokenizing Character Set

- + [defaultTokenizingCharacterSet](#) (page 7)  
Returns the default tokenizing character set.
- [setTokenizingCharacterSet:](#) (page 9)  
Sets the receiver's tokenizing character set to a given character set.

- [tokenizingCharacterSet](#) (page 10)  
Returns the receiver's tokenizing character set.

## Configuring the Completion Delay

- [setCompletionDelay:](#) (page 8)  
Sets the receiver's completion delay to a given delay.
- [completionDelay](#) (page 8)  
Returns the receiver's completion delay.
- + [defaultCompletionDelay](#) (page 7)  
Returns the default completion delay.

## Managing the Delegate

- [delegate](#) (page 8)  
Returns the receiver's delegate.
- [setDelegate:](#) (page 9)  
Sets the receiver's delegate.

## Displaying Tokenized Strings

- [textFieldCell:displayStringForRepresentedObject:](#) (page 11) *delegate method*  
Allows the delegate to provide a string to be displayed as a proxy for the represented object.
- [textFieldCell:styleForRepresentedObject:](#) (page 14) *delegate method*  
Allows the delegate to return the token style for editing the specified represented object.

## Editing Tokenized Strings

- [textFieldCell:editingStringForRepresentedObject:](#) (page 12) *delegate method*  
Allows the delegate to provide a string to be edited as a proxy for the represented object.
- [textFieldCell:completionsForSubstring:indexOfToken:indexOfSelectedItem:](#) (page 11) *delegate method*  
Allows the delegate to provide an array of appropriate completions for the contents of the receiver.
- [textFieldCell:representedObjectForEditingString:](#) (page 13) *delegate method*  
Allows the delegate to provide a represented object for the string being edited.
- [textFieldCell:shouldAddObjects:atIndex:](#) (page 13) *delegate method*  
Allows the delegate to validate the tokens to be added to the receiver at a given index.

## Reading From and Writing to the Pasteboard

- [textFieldCell:readFromPasteboard:](#) (page 12) *delegate method*  
Allows the delegate to return an array of objects representing the data read from *pboard*.

- `tokenFieldCell:writeRepresentedObjects:toPasteboard:` (page 14) *delegate method*  
Allows the delegate the opportunity to write custom pasteboard types to the pasteboard for the represented objects in *objects*.

## Managing Menus for Represented Objects

- `tokenFieldCell:hasMenuForRepresentedObject:` (page 12) *delegate method*  
Allows the delegate to specify whether the represented object provides a menu.
- `tokenFieldCell:menuForRepresentedObject:` (page 12) *delegate method*  
Allows the delegate to provide a menu for the specified represented object.

## Class Methods

### defaultCompletionDelay

Returns the default completion delay.

```
+ (NSTimeInterval)defaultCompletionDelay
```

#### Return Value

The default completion delay.

#### Discussion

The default completion delay is 0.

#### Availability

Available in Mac OS X v10.4 and later.

#### Declared In

NSTokenFieldCell.h

### defaultTokenizingCharacterSet

Returns the default tokenizing character set.

```
+ (NSCharacterSet *)defaultTokenizingCharacterSet
```

#### Return Value

The default tokenizing character set.

#### Discussion

The default tokenizing character set contains the single character “,”.

#### Availability

Available in Mac OS X v10.4 and later.

#### Declared In

NSTokenFieldCell.h

## Instance Methods

### completionDelay

Returns the receiver's completion delay.

- (NSTimeInterval)completionDelay

#### Return Value

The receiver's completion delay.

#### Availability

Available in Mac OS X v10.4 and later.

#### See Also

- [setCompletionDelay:](#) (page 8)

+ [defaultCompletionDelay](#) (page 7)

#### Declared In

NSTextFieldCell.h

### delegate

Returns the receiver's delegate.

- (id)delegate

#### Return Value

The receiver's delegate.

#### Availability

Available in Mac OS X v10.4 and later.

#### See Also

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

#### Declared In

NSTextFieldCell.h

### setCompletionDelay:

Sets the receiver's completion delay to a given delay.

- (void)setCompletionDelay:(NSTimeInterval)delay

#### Parameters

*delay*

The delay for the receiver.

#### Availability

Available in Mac OS X v10.4 and later.

#### See Also

- [completionDelay](#) (page 8)

#### Declared In

NSTextFieldCell.h

### setDelegate:

Sets the receiver's delegate.

- (void)setDelegate:(id)*anObject*

#### Parameters

*anObject*

The delegate for the receiver.

#### Availability

Available in Mac OS X v10.4 and later.

#### See Also

- [delegate](#) (page 8)

#### Declared In

NSTextFieldCell.h

### setTokenizingCharacterSet:

Sets the receiver's tokenizing character set to a given character set.

- (void)setTokenizingCharacterSet:(NSCharacterSet \*)*characterSet*

#### Parameters

*characterSet*

The tokenizing character set for the receiver.

#### Availability

Available in Mac OS X v10.4 and later.

#### See Also

- [tokenizingCharacterSet](#) (page 10)

#### Declared In

NSTextFieldCell.h

### setTokenStyle:

Sets the token style of the receiver.

- (void)setTokenStyle:(NSTokenStyle)*style*

**Parameters***style*

The token style for the receiver.

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

- [tokenStyle](#) (page 10)

**Declared In**

NSTextFieldCell.h

## tokenizingCharacterSet

Returns the receiver's tokenizing character set.

- (NSCharacterSet \*)tokenizingCharacterSet

**Return Value**

The receiver's tokenizing character set.

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

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

+ [defaultTokenizingCharacterSet](#) (page 7)

**Declared In**

NSTextFieldCell.h

## tokenStyle

Returns the receiver's token style.

- (NSTokenStyle)tokenStyle

**Return Value**

The receiver's token style.

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

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

**Declared In**

NSTextFieldCell.h

## Delegate Methods

### **tokenFieldCell:completionsForSubstring:indexOfToken:indexOfSelectedItem:**

Allows the delegate to provide an array of appropriate completions for the contents of the receiver.

```
- (NSArray *)tokenFieldCell:(NSTokenFieldCell *)tokenFieldCell
    completionsForSubstring:(NSString *)substring indexOfToken:(NSInteger)tokenIndex
    indexOfSelectedItem:(NSInteger *)selectedIndex
```

#### **Parameters**

*tokenFieldCell*

The token field cell sending the message to the delegate.

*substring*

The partial string that to be completed.

*tokenIndex*

The index of the token being edited.

*selectedIndex*

You may return by reference an index in the returned array that specifies which of the completions should be initially selected. If you do not want an initial selection, return `NULL`.

#### **Return Value**

An array of strings for completing *substring*.

#### **Discussion**

The default behavior provides no completions.

#### **Availability**

Available in Mac OS X v10.4 and later.

#### **Declared In**

`NSTokenFieldCell.h`

### **tokenFieldCell:displayStringForRepresentedObject:**

Allows the delegate to provide a string to be displayed as a proxy for the represented object.

```
- (NSString *)tokenFieldCell:(NSTokenFieldCell *)tokenFieldCell
    displayStringForRepresentedObject:(id)representedObject
```

#### **Discussion**

If you return `nil` or do not implement this method, then *representedObject* is displayed as the string.

#### **Availability**

Available in Mac OS X v10.4 and later.

#### **Declared In**

`NSTokenFieldCell.h`

**textFieldCell:editingStringForRepresentedObject:**

Allows the delegate to provide a string to be edited as a proxy for the represented object.

```
- (NSString *)textFieldCell:(NSTextFieldCell *)textFieldCell
    editingStringForRepresentedObject:(id)representedObject
```

**Discussion**

If you return `nil` or do not implement this method, then the value returned by [textFieldCell:displayStringForRepresentedObject:](#) (page 11) is used for editing.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTextFieldCell.h

**textFieldCell:hasMenuForRepresentedObject:**

Allows the delegate to specify whether the represented object provides a menu.

```
- (BOOL)textFieldCell:(NSTextFieldCell *)textFieldCell
    hasMenuForRepresentedObject:(id)representedObject
```

**Discussion**

By default tokens have no menus.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTextFieldCell.h

**textFieldCell:menuForRepresentedObject:**

Allows the delegate to provide a menu for the specified represented object.

```
- (NSMenu *)textFieldCell:(NSTextFieldCell *)textFieldCell
    menuForRepresentedObject:(id)representedObject
```

**Discussion**

The returned menu should be autoreleased. By default tokens do not return menus.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTextFieldCell.h

**textFieldCell:readFromPasteboard:**

Allows the delegate to return an array of objects representing the data read from *pboard*.

```
- (NSArray *)tokenFieldCell:(NSTextFieldCell *)tokenFieldCell
  readFromPasteboard:(NSPasteboard *)pboard
```

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTextFieldCell.h

**tokenFieldCell:representedObjectForEditingString:**

Allows the delegate to provide a represented object for the string being edited.

```
- (id)tokenFieldCell:(NSTextFieldCell *)tokenFieldCell
  representedObjectForEditingString:(NSString *)editingString
```

**Parameters**

*tokenFieldCell*

The token field cell sending the message to the delegate.

*editingString*

The string being edited.

**Return Value**

A represented object associated with *editingString*.

**Discussion**

If your application uses some object other than an `NSString` object for their represented objects, you should return a new, autoreleased instance of that object from this method.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTextFieldCell.h

**tokenFieldCell:shouldAddObjectsAtIndex:**

Allows the delegate to validate the tokens to be added to the receiver at a given index.

```
- (NSArray *)tokenFieldCell:(NSTextFieldCell *)tokenFieldCell
  shouldAddObjects:(NSArray *)tokens atIndex:(NSUInteger)index
```

**Parameters**

*tokenFieldCell*

The token field cell sending the message to the delegate.

*tokens*

An array of tokens to be added to the receiver.

*index*

The index in the receiver where the tokens are to be added.

**Return Value**

An array of validated tokens.

**Discussion**

The delegate can return the array unchanged or return a modified array of tokens. To reject the add completely, return an empty array. Returning `nil` causes an error.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

`NSTokenFieldCell.h`

**tokenFieldCell:styleForRepresentedObject:**

Allows the delegate to return the token style for editing the specified represented object.

```
- (NSTokenStyle)tokenFieldCell:(NSTokenFieldCell *)tokenFieldCell
    styleForRepresentedObject:(id)representedObject
```

**Discussion**

The delegate should return `NSDefaultTokenStyle`, `NSPlainTextTokenStyle` or `NSRoundedTokenStyle`.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

`NSTokenFieldCell.h`

**tokenFieldCell:writeRepresentedObjects:toPasteboard:**

Allows the delegate the opportunity to write custom pasteboard types to the pasteboard for the represented objects in *objects*.

```
- (BOOL)tokenFieldCell:(NSTokenFieldCell *)tokenFieldCell
    writeRepresentedObjects:(NSArray *)objects toPasteboard:(NSPasteboard *)pboard
```

**Discussion**

The display strings for the represented objects have already been placed on the pasteboard as both `NSStringPboardType` and an array of `NSString` objects.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

`NSTokenFieldCell.h`

## Constants

**NSTokenStyle**

The `NSTokenStyle` type provides storage for values defined in “`NSTokenStyle Values`”.

```
typedef NSUInteger NSTokenStyle;
```

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTokenFieldCell.h

**NSTokenStyle Values**

The NSTokenStyle constants define how tokens are displayed and editable in the NSTokenFieldCell. These values are used by [tokenStyle](#) (page 10), [setTokenStyle:](#) (page 9) and the delegate method [tokenFieldCell:styleForRepresentedObject:](#) (page 14).

```
enum {
    NSDefaultTokenStyle,
    NSPlainTextTokenStyle,
    NSRoundedTokenStyle
};
```

**Constants**

NSDefaultTokenStyle

Style best used for keyword type tokens.

Available in Mac OS X v10.4 and later.

Declared in NSTokenFieldCell.h.

NSPlainTextTokenStyle

Style to use for data you want represented as plain-text and without any token background.

Available in Mac OS X v10.4 and later.

Declared in NSTokenFieldCell.h.

NSRoundedTokenStyle

Style best used for address type tokens.

Available in Mac OS X v10.4 and later.

Declared in NSTokenFieldCell.h.

**Declared In**

NSTokenFieldCell.h



# Document Revision History

---

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

Date	Notes
2007-03-05	Update with Mac OS X v10.5 API.
2007-02-08	Corrected the declaration of the NSTokenStyle enumeration.
2006-11-07	Corrected description of setTokenStyle:.
2006-05-23	First publication of this content as a separate document.

## REVISION HISTORY

### Document Revision History

# Index

---

## C

---

completionDelay **instance method** [8](#)

## D

---

defaultCompletionDelay **class method** [7](#)  
defaultTokenizingCharacterSet **class method** [7](#)  
delegate **instance method** [8](#)

## N

---

NSDefaultTokenStyle **constant** [15](#)  
NSPlainTextTokenStyle **constant** [15](#)  
NSRoundedTokenStyle **constant** [15](#)  
NSTokenStyle **data type** [14](#)  
NSTokenStyle Values [15](#)

## S

---

setCompletionDelay: **instance method** [8](#)  
setDelegate: **instance method** [9](#)  
setTokenizingCharacterSet: **instance method** [9](#)  
setTokenStyle: **instance method** [9](#)

## T

---

tokenFieldCell:completionsForSubstring:  
    indexOfToken:indexOfSelectedItem:  
    <NSObject> **delegate method** [11](#)  
tokenFieldCell:displayStringForRepresentedObject:  
    <NSObject> **delegate method** [11](#)  
tokenFieldCell:editingStringForRepresentedObject:  
    <NSObject> **delegate method** [12](#)

tokenFieldCell:hasMenuForRepresentedObject:  
    <NSObject> **delegate method** [12](#)  
tokenFieldCell:menuForRepresentedObject:  
    <NSObject> **delegate method** [12](#)  
tokenFieldCell:readFromPasteboard: <NSObject>  
    **delegate method** [12](#)  
tokenFieldCell:representedObjectForEditingString:  
    <NSObject> **delegate method** [13](#)  
tokenFieldCell:shouldAddObjects:atIndex:  
    <NSObject> **delegate method** [13](#)  
tokenFieldCell:styleForRepresentedObject:  
    <NSObject> **delegate method** [14](#)  
tokenFieldCell:writeRepresentedObjects:  
    toPasteboard: <NSObject> **delegate method** [14](#)  
tokenizingCharacterSet **instance method** [10](#)  
tokenStyle **instance method** [10](#)