

---

# NSTextField Class Reference

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



2008-10-15



Apple Inc.  
© 2008 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

---

## **NSTextField Class Reference 5**

---

|   |    |
|---|----|
| Overview  | 5  |
| Tasks   | 6  |
| Configuring the Token Style   | 6  |
| Configuring the Tokenizing Character Set                            | 6  |
| Configuring the Completion Delay                                    | 6  |
| Displaying a Tokenized String                                       | 6  |
| Editing a Tokenized String  | 7  |
| Reading To and Writing From the Pasteboard                          | 7  |
| Providing a Menu  | 7  |
| Class Methods   | 7  |
| defaultCompletionDelay  | 7  |
| defaultTokenizingCharacterSet                                       | 8  |
| Instance Methods  | 8  |
| completionDelay   | 8  |
| setCompletionDelay:   | 8  |
| setTokenizingCharacterSet:  | 9  |
| setTokenStyle:  | 9  |
| tokenizingCharacterSet  | 9  |
| tokenStyle  | 9  |
| Delegate Methods  | 10 |
| textField:completionsForSubstring:indexOfToken:indexOfSelectedItem: | 10 |
| textField:displayStringForRepresentedObject:                        | 10 |
| textField:editingStringForRepresentedObject:                        | 11 |
| textField:hasMenuForRepresentedObject:                              | 11 |
| textField:menuForRepresentedObject:                                 | 12 |
| textField:readFromPasteboard:                                       | 12 |
| textField:representedObjectForEditingString:                        | 12 |
| textField:shouldAddObjects:atIndex:                                 | 13 |
| textField:styleForRepresentedObject:                                | 13 |
| textField:writeRepresentedObjects:toPasteboard:                     | 14 |

---

## **Document Revision History 15**

---

## **Index 17**

---



# NSTextField Class Reference

---

|                        |   |
|------------------------|---|
| <b>Inherits from</b>   | NSTextField : NSControl : NSView : NSResponder : NSObject   |
| <b>Conforms to</b>     | NSUserInterfaceValidations (NSTextField)<br>NSAnimatablePropertyContainer (NSView)<br>NSCoding (NSResponder)<br>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>     | NSTextField.h   |

## Overview

`NSTextField` is a subclass of `NSTextField` that provides tokenized editing similar to the address field in the Mail application.

`NSTextField` uses an `NSTextFieldCell` to implement much of the control's functionality. `NSTextField` provides cover methods for most methods of `NSTextFieldCell`, which invoke the corresponding cell method.

**Notes:** In Mac OS X v10.4 and earlier, represented objects associated with token fields had to conform to `NSCoding`. Starting with Mac OS X v10.5, they no longer need to.

In Mac OS X v10.4, `NSTextField` trims whitespace around tokens but it does not trim whitespace in Mac OS X versions 10.5.0 and 10.5.1. In Mac OS X v10.5.2, you get whitespace-trimming behavior by either linking against the v10.4 binary or linking against the v10.5 binary and *not* implementing the `textField:representedObjectForEditingString:` (page 12) method. If you do not want the whitespace-trimming behavior, link against the v10.5 binary and implement this method, returning the editing string if you have no represented object.

## Tasks

### Configuring the Token Style

- `setTokenStyle:` (page 9)  
Returns the token style of the receiver.
- `tokenStyle` (page 9)  
Returns the receiver's token style.

### Configuring the Tokenizing Character Set

- `setTokenizingCharacterSet:` (page 9)  
Sets the receiver's tokenizing character set to *characterSet*.
- `tokenizingCharacterSet` (page 9)  
Returns the receiver's tokenizing character set.
- + `defaultTokenizingCharacterSet` (page 8)  
Returns the default tokenizing character set.

### Configuring the Completion Delay

- `setCompletionDelay:` (page 8)  
Sets the receiver's completion delay.
- `completionDelay` (page 8)  
Returns the receiver's completion delay.
- + `defaultCompletionDelay` (page 7)  
Returns the default completion delay.

### Displaying a Tokenized String

- `textField:displayStringForRepresentedObject:` (page 10) *delegate method*  
Allows the delegate to provide a string to be displayed as a proxy for the given represented object.

- `textField:styleForRepresentedObject:` (page 13) *delegate method*  
Allows the delegate to return the token style for editing the specified represented object.

## Editing a Tokenized String

- `textField:completionsForSubstring:indexOfToken:indexOfSelectedItem:` (page 10) *delegate method*  
Allows the delegate to provide an array of appropriate completions for the contents of the receiver.
- `textField:editingStringForRepresentedObject:` (page 11) *delegate method*  
Allows the delegate to provide a string to be edited as a proxy for a represented object.
- `textField:representedObjectForEditingString:` (page 12) *delegate method*  
Allows the delegate to provide a represented object for the given editing string.
- `textField:shouldAddObjects:atIndex:` (page 13) *delegate method*  
Allows the delegate to validate the tokens to be added to the receiver at a particular location.

## Reading To and Writing From the Pasteboard

- `textField:readFromPasteboard:` (page 12) *delegate method*  
Allows the delegate to return an array of objects representing the data read from the specified pasteboard.
- `textField:writeRepresentedObjects:toPasteboard:` (page 14) *delegate method*  
Sent so the delegate can write represented objects to the pasteboard corresponding to a given array of display strings.

## Providing a Menu

- `textField:hasMenuForRepresentedObject:` (page 11) *delegate method*  
Allows the delegate to specify whether the given represented object provides a menu.
- `textField: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
```

#### **Discussion**

The default completion delay is 0.

#### **Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTokenField.h

**defaultTokenizingCharacterSet**

Returns the default tokenizing character set.

+ (NSCharacterSet \*)defaultTokenizingCharacterSet

**Discussion**

The default tokenizing character set is “,”.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTokenField.h

## Instance Methods

**completionDelay**

Returns the receiver's completion delay.

- (NSTimeInterval)completionDelay

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**- [setCompletionDelay:](#) (page 8)+ [defaultCompletionDelay](#) (page 7)**Declared In**

NSTokenField.h

**setCompletionDelay:**

Sets the receiver's completion delay.

- (void)setCompletionDelay:(NSTimeInterval)delay

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**- [completionDelay](#) (page 8)**Declared In**

NSTokenField.h



## setTokenizingCharacterSet:

Sets the receiver's tokenizing character set to *characterSet*.

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

### Availability

Available in Mac OS X v10.4 and later.

### See Also

- [tokenizingCharacterSet](#) (page 9)

### Declared In

NSTextField.h

## setTokenStyle:

Returns the token style of the receiver.

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

### Availability

Available in Mac OS X v10.4 and later.

### See Also

- [tokenStyle](#) (page 9)

### Declared In

NSTextField.h

## tokenizingCharacterSet

Returns the receiver's tokenizing character set.

- (NSCharacterSet \*)tokenizingCharacterSet

### Availability

Available in Mac OS X v10.4 and later.

### See Also

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

+ [defaultTokenizingCharacterSet](#) (page 8)

### Declared In

NSTextField.h

## tokenStyle

Returns the receiver's token style.

- (NSTokenStyle)tokenStyle

**Availability**

Available in Mac OS X v10.4 and later.

**See Also**

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

**Declared In**

NSTextField.h

## Delegate Methods

### **tokenField:completionsForSubstring:indexOfToken:indexOfSelectedItem:**

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

```
- (NSArray *)tokenField:(NSTextField *)tokenField completionsForSubstring:(NSString *)substring indexOfToken:(NSInteger)tokenIndex indexOfSelectedItem:(NSInteger *)selectedIndex
```

**Parameters**

*tokenField*

The token field where editing is occurring.

*substring*

The partial string that is to be completed.

*tokenIndex*

The index of the token being edited.

*selectedIndex*

Optionally, you can return by-reference an index into the returned array that specifies which of the completions should be initially selected. If none are to be selected, return by reference -1.

**Return Value**

An array of possible completions.

**Discussion**

The default behavior of this method provides no completions.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTextField.h

### **tokenField:displayStringForRepresentedObject:**

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

```
- (NSString *)tokenField:(NSTextField *)tokenField displayStringForRepresentedObject:(id)representedObject
```

**Parameters***tokenField*

The token field.

*representedObject*

A represented object of the token field.

**Return Value**

The string to be used as a proxy for `representedObject`. 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**

NSTextField.h

**tokenField:editingStringForRepresentedObject:**

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

```
- (NSString *)tokenField:(NSTextField *)tokenField
    editingStringForRepresentedObject:(id)representedObject
```

**Discussion**

If you return `nil` from this method, the token field is not editable.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTextField.h

**tokenField:hasMenuForRepresentedObject:**

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

```
- (BOOL)tokenField:(NSTextField *)tokenField
    hasMenuForRepresentedObject:(id)representedObject
```

**Parameters***tokenField*

The token field.

*representedObject*

A represented object of the token field.

**Return Value**

YES if the represented object has a menu, NO otherwise.

**Discussion**

By default tokens have no menus.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTextField.h

**tokenField:menuForRepresentedObject:**

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

```
- (NSMenu *)tokenField:(NSTextField *)tokenField
  menuForRepresentedObject:(id)representedObject
```

**Parameters***tokenField*

The token field.

*representedObject*

A represented object of the token field.

**Return Value**

The menu associated with the represented object.

**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**

NSTextField.h

**tokenField:readFromPasteboard:**

Allows the delegate to return an array of objects representing the data read from the specified pasteboard.

```
- (NSArray *)tokenField:(NSTextField *)tokenField readFromPasteboard:(NSPasteboard
  *)pboard
```

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTextField.h

**tokenField:representedObjectForEditingString:**

Allows the delegate to provide a represented object for the given editing string.

```
- (id)tokenField:(NSTextField *)tokenField
  representedObjectForEditingString:(NSString *)editingString
```

**Discussion**

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

**Note:** In Mac OS X v10.4, `NSTokenField` trims whitespace around tokens but it does not trim whitespace in Mac OS X versions 10.5.0 and 10.5.1. In Mac OS X v10.5.2, you get whitespace-trimming behavior by either linking against the v10.4 binary or linking against the v10.5 binary and *not* implementing the this method. If you do not want the whitespace-trimming behavior, link against the v10.5 binary and implement this method, returning the editing string if you have no represented object.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

`NSTokenField.h`

**tokenField:shouldAddObjectsAtIndex:**

Allows the delegate to validate the tokens to be added to the receiver at a particular location.

```
- (NSArray *)tokenField:(NSTokenField *)tokenField shouldAddObjects:(NSArray *)tokens
  atIndex:(NSUInteger)index
```

**Parameters**

*tokenField*

The token field being validated.

*tokens*

An array of tokens to be inserted in the receiver at `index`.

*index*

The index of the receiver in which the array of tokens to be validated (`tokens`) will be inserted.

**Return Value**

The array of tokens to be inserted.

**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**

`NSTokenField.h`

**tokenField:styleForRepresentedObject:**

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

```
- (NSTokenStyle)tokenField:(NSTokenField *)tokenField
  styleForRepresentedObject:(id)representedObject
```

**Discussion**

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

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTokenField.h

**tokenField:writeRepresentedObjects:toPasteboard:**

Sent so the delegate can write represented objects to the pasteboard corresponding to a given array of display strings.

```
- (BOOL)tokenField:(NSTokenField *)tokenField writeRepresentedObjects:(NSArray *)objects toPasteboard:(NSPasteboard *)pboard
```

**Parameters**

*tokenField*

The token field.

*objects*

An array of represented objects associated with the token field.

*pboard*

The pasteboard to which to write the represented objects.

**Return Value**

YES if the delegate writes the represented objects to the pasteboard, NO otherwise. If NO, the token field writes the display strings to the NSStringPboardType pasteboard.

**Availability**

Available in Mac OS X v10.4 and later.

**Declared In**

NSTokenField.h

# Document Revision History

---

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

| Date       | Notes   |
|------------|---|
| 2008-10-15 | Made several minor corrections.                           |
| 2007-03-07 | Formatting changes. Revised task headings.                |
| 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 8**

## S

---

setCompletionDelay: **instance method 8**

setTokenizingCharacterSet: **instance method 9**

setTokenStyle: **instance method 9**

## T

---

tokenField:completionsForSubstring:indexOfToken:  
    indexOfSelectedItem: **<NSObject> delegate  
    method 10**

tokenField:displayStringForRepresentedObject:  
    **<NSObject> delegate method 10**

tokenField:editingStringForRepresentedObject:  
    **<NSObject> delegate method 11**

tokenField:hasMenuForRepresentedObject:  
    **<NSObject> delegate method 11**

tokenField:menuForRepresentedObject:  
    **<NSObject> delegate method 12**

tokenField:readFromPasteboard: **<NSObject>  
    delegate method 12**

tokenField:representedObjectForEditingString:  
    **<NSObject> delegate method 12**

tokenField:shouldAddObjects:atIndex:  
    **<NSObject> delegate method 13**

tokenField:styleForRepresentedObject:  
    **<NSObject> delegate method 13**

tokenField:writeRepresentedObjects:toPasteboard:

**<NSObject> delegate method 14**

tokenizingCharacterSet **instance method 9**

tokenStyle **instance method 9**