

---

# Core Text String Attributes Reference

[Carbon](#) > [Text & Fonts](#)



2007-05-01



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, Carbon, Mac, and Mac OS are trademarks of Apple Inc., registered in the United States and other countries.

Helvetica is a registered trademark of Heidelberger Druckmaschinen AG, available from Linotype Library GmbH.

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

---

## **Core Text String Attributes Reference 5**

---

Overview 5

Constants 5

String Attribute Name Constants 5

CTUnderlineStyle 7

CTUnderlineStyleModifiers 7

---

## **Document Revision History 9**

---

---

## **Index 11**

---



# Core Text String Attributes Reference

---

<b>Framework:</b>	ApplicationServices/CoreText
<b>Declared in</b>	CTStringAttributes.h

## Overview

This reference document describes the attributes to which Core Text responds when the attributes are placed in a `CFAttributedString` object.

## Constants

### String Attribute Name Constants

These constants represent string attribute names.

```
const CFStringRef kCTFontAttributeName;  
const CFStringRef kCTKernAttributeName;  
const CFStringRef kCTLigatureAttributeName;  
const CFStringRef kCTForegroundColorAttributeName;  
const CFStringRef kCTParagraphStyleAttributeName;  
const CFStringRef kCTUnderlineStyleAttributeName;  
const CFStringRef kCTVerticalFormsAttributeName;  
const CFStringRef kCTGlyphInfoAttributeName;
```

#### Constants

`kCTFontAttributeName`

The font of the text to which this attribute applies. The value associated with this attribute must be a `CTFont` object. Default is Helvetica 12.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

`kCTKernAttributeName`

The amount to kern the next character. The value associated with this attribute must be a `CGFloat`. Default is standard kerning. The kerning attribute indicates how many points the following character should be shifted from its default offset as defined by the current character's font in points: a positive kern indicates a shift farther away from and a negative kern indicates a shift closer to the current character. If this attribute is not present, standard kerning is used. If this attribute is set to `0.0`, no kerning is done at all.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

**kCTLigatureAttributeName**

The type of ligatures to use. The value associated with this attribute must be a `CFNumber` object. Default is an integer value of 1. The ligature attribute determines what kinds of ligatures should be used when displaying the string. A value of 0 indicates that only ligatures essential for proper rendering of text should be used. A value of 1 indicates that standard ligatures should be used, and 2 indicates that all available ligatures should be used. Which ligatures are standard depends on the script and possibly the font. Arabic text, for example, requires ligatures for many character sequences but has a rich set of additional ligatures that combine characters. English text has no essential ligatures, and typically has only two standard ligatures, those for "fi" and "fl"—all others are considered more advanced or fancy.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

**kCTForegroundColorAttributeName**

The foreground color of the text to which this attribute applies. The value associated with this attribute must be a `CGColor` object. Default value is black.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

**kCTParagraphStyleAttributeName**

The paragraph style of the text to which this attribute applies. A paragraph style object is used to specify things like line alignment, tab rulers, writing direction, and so on. Value must be a `CTParagraphStyle` object. Default is an empty `CTParagraphStyle` object. See *CTParagraphStyle Reference* for more information.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

**kCTUnderlineStyleAttributeName**

The style of underlining, to be applied at render time, for the text to which this attribute applies. Value must be a `CFNumber` object. Default is `kCTUnderlineStyleNone`. Set a value of something other than `kCTUnderlineStyleNone` to draw an underline. In addition, the constants listed in [“CTUnderlineStyleModifiers”](#) (page 7) can be used to modify the look of the underline. The underline color is determined by the text's foreground color.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

**kCTVerticalFormsAttributeName**

The orientation of the glyphs in the text to which this attribute applies. Value must be a `CFBoolean` object. Default is `False`. A value of `False` indicates that horizontal glyph forms are to be used; `True` indicates that vertical glyph forms are to be used.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

**kCTGlyphInfoAttributeName**

The glyph info object to apply to the text associated with this attribute. Value must be a `CTGlyphInfo` object. The glyph specified by this `CTGlyphInfo` object is assigned to the entire attribute range, provided that its contents match the specified base string and that the specified glyph is available in the font specified by `kCTFontAttributeName`. See *CTGlyphInfo Reference* for more information.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

**Declared In**

`CTStringAttributes.h`

## CTUnderlineStyle

Underline style specifiers.

```
enum{
kCTUnderlineStyleNone = 0x00,
kCTUnderlineStyleSingle = 0x01,
kCTUnderlineStyleThick = 0x02,
kCTUnderlineStyleDouble = 0x09
};
typedef int32_t CTUnderlineStyle;
```

### Constants

kCTUnderlineStyleNone

Do not draw an underline.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

kCTUnderlineStyleSingle

Draw an underline consisting of a single line.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

kCTUnderlineStyleThick

Draw an underline consisting of a thick line.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

kCTUnderlineStyleDouble

Draw an underline consisting of a double line.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

### Discussion

These underline type specifiers can be applied to the value set with the [kCTUnderlineStyleAttributeName](#) (page 6) attribute to control the underline style Core Text uses when rendering the text to which the attribute applies.

### Declared In

`CTStringAttributes.h`

## CTUnderlineStyleModifiers

Underline style modifiers.

```
enum{
kCTUnderlinePatternSolid = 0x0000,
kCTUnderlinePatternDot = 0x0100,
kCTUnderlinePatternDash = 0x0200,
kCTUnderlinePatternDashDot = 0x0300,
kCTUnderlinePatternDashDotDot = 0x0400
};
typedef int32_t CTUnderlineStyleModifiers;
```

**Constants**

`kCTUnderlinePatternSolid`

Draw a solid underline.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

`kCTUnderlinePatternDot`

Draw an underline using a pattern of dots.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

`kCTUnderlinePatternDash`

Draw an underline using a pattern of dashes.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

`kCTUnderlinePatternDashDot`

Draw an underline using a pattern of alternating dashes and dots.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

`kCTUnderlinePatternDashDotDot`

Draw an underline using a pattern of a dash followed by two dots.

Available in Mac OS X v10.5 and later.

Declared in `CTStringAttributes.h`.

**Discussion**

Set these bits with the underline style (see [“CTUnderlineStyle”](#) (page 7)) that you set with the [`kCTUnderlineStyleAttributeName`](#) (page 6) attribute to modify how the underline will be drawn.

**Declared In**

`CTStringAttributes.h`



# Document Revision History

---

This table describes the changes to *Core Text String Attributes Reference*.

Date	Notes
2007-05-01	New document that describes the NSAttributedString attributes recognized by Core Text.

## REVISION HISTORY

### Document Revision History

# Index

---

## C

---

CTUnderlineStyle [7](#)  
CTUnderlineStyleModifiers [7](#)

## K

---

kCTFontAttributeName [constant 5](#)  
kCTForegroundColorAttributeName [constant 6](#)  
kCTGlyphInfoAttributeName [constant 6](#)  
kCTKernAttributeName [constant 5](#)  
kCTLigatureAttributeName [constant 6](#)  
kCTParagraphStyleAttributeName [constant 6](#)  
kCTUnderlinePatternDash [constant 8](#)  
kCTUnderlinePatternDashDot [constant 8](#)  
kCTUnderlinePatternDashDotDot [constant 8](#)  
kCTUnderlinePatternDot [constant 8](#)  
kCTUnderlinePatternSolid [constant 8](#)  
kCTUnderlineStyleAttributeName [constant 6](#)  
kCTUnderlineStyleDouble [constant 7](#)  
kCTUnderlineStyleNone [constant 7](#)  
kCTUnderlineStyleSingle [constant 7](#)  
kCTUnderlineStyleThick [constant 7](#)  
kCTVerticalFormsAttributeName [constant 6](#)

## S

---

String Attribute Name Constants [5](#)