


NSParagraphStyle

Inherits From:	NSObject
Conforms To:	NSCoding NSCopying NSMutableCopying NSObject (NSObject)
Declared In:	UIKit/NSParagraphStyle.h

Class Description

NSParagraphStyle and its subclass NSMutableParagraphStyle encapsulate the paragraph or ruler attributes used by the NSAttributedString classes. Instances of these classes are often referred to as *paragraph style objects*, or when no confusion will result, as *paragraph styles*.

A paragraph style object represents a complex attribute value in an attributed string, storing a number of sub-attributes that affect paragraph layout for the characters of the string. Among these sub-attributes are alignment, tab stops, and indents. See the method descriptions for more information on each sub-attribute.

Adopted Protocols

NSCoding	– encodeWithCoder: – initWithCoder:
NSCopying	– copyWithZone:
NSMutableCopying	– mutableCopyWithZone:

Method Types

Creating an NSParagraphStyle	+ defaultParagraphStyle
------------------------------	-------------------------

Accessing style information

- alignment
- firstLineHeadIndent
- headIndent
- tailIndent
- tabStops
- lineBreakMode
- maximumLineHeight
- minimumLineHeight
- lineSpacing
- paragraphSpacing

Class Methods

defaultParagraphStyle

+ (NSMutableParagraphStyle *)defaultParagraphStyle

Returns the default paragraph style, which contains these values:

Sub-Attribute	Default Value
Alignment	NSNaturalTextAlignment
Tab stops	12 left-aligned tabs, spaced by 28.0 points
Line break mode	NSLineBreakByWordWrapping
All others	0.0

See individual method descriptions for explanations of each sub-attribute.

See also: – `initWithAttributes:`

Instance Methods

alignment

– (NSTextAlignment)alignment

Returns the text alignment of the paragraph style, one of:

- NSLeftTextAlignment
- NSRightTextAlignment
- NSCenterTextAlignment
- NSJustifiedTextAlignment
- NSNaturalTextAlignment

Natural text alignment is realized as left or right alignment depending on the line sweep direction of the first script contained in the paragraph.

See also: – **setAlignment:** (NSMutableParagraphStyle)

firstLineHeadIndent

– (float)**firstLineHeadIndent**

Returns the distance in points from the leading margin of a text container to the beginning of the paragraph's first line. This value is always nonnegative.

See also: – **headIndent**, – **tailIndent**, – **setFirstLineHeadIndent:** (NSMutableParagraphStyle)

headIndent

– (float)**headIndent**

Returns the distance in points from the leading margin of a text container to the beginning of lines other than the first. This value is always nonnegative.

See also: – **firstLineHeadIndent**, – **tailIndent**, – **setHeadIndent:** (NSMutableParagraphStyle)

lineBreakMode

– (NSLineBreakMode)**lineBreakMode**

Returns the mode that should be used to break lines when laying out the paragraph's text. This is one of:

Value	Meaning
NSLineBreakByWordWrapping	Wrapping occurs at word boundaries, unless the word itself doesn't fit on a single line.
NSLineBreakByCharWrapping	Wrapping occurs before the first character that doesn't fit.
NSLineBreakByClipping	Lines are simply not drawn past the edge of the text container.
NSLineBreakByTruncatingHead	Each line is displayed so that the end fits in the container and the missing text is indicated by some kind of ellipsis glyph.
NSLineBreakByTruncatingTail	Each line is displayed so that the beginning fits in the container and the missing text is indicated by some kind of ellipsis glyph.
NSLineBreakByTruncatingMiddle	Each line is displayed so that the beginning and end fit in the container and the missing text is indicated by some kind of ellipsis glyph in the middle.

See also: – **setLineBreakMode:** (NSMutableParagraphStyle)

lineSpacing

– (float)**lineSpacing**

Returns the space in points added between lines within the paragraph (commonly known as *leading*). This value is always nonnegative.

See also: – **maximumLineHeight**, – **minimumLineHeight**, – **paragraphSpacing**,
– **setLineSpacing:** (NSMutableParagraphStyle)

maximumLineHeight

– (float)**maximumLineHeight**

Returns the maximum height that any line in the paragraph style will occupy, regardless of the font size or size of any attached graphic. Glyphs and graphics exceeding this height will overlap neighboring lines; however, a maximum height of zero implies no line height limit. This value is always nonnegative. The default value is zero.

Note: Although this limit applies to the line itself, line spacing adds extra space between adjacent lines.

See also: – **minimumLineHeight**, – **lineSpacing**,
– **setMaximumLineHeight:** (NSMutableParagraphStyle)

minimumLineHeight

– (float)**minimumLineHeight**

Returns the minimum height that any line in the paragraph style will occupy, regardless of the font size or size of any attached graphic. This value is always nonnegative.

See also: – **maximumLineHeight**, – **lineSpacing**,
– **setMinimumLineHeight:** (NSMutableParagraphStyle)

paragraphSpacing

– (float)**paragraphSpacing**

Returns the space added at the end of the paragraph to separate it from the following paragraph. This value is always nonnegative.

See also: – **lineSpacing**, – **setParagraphSpacing:** (NSMutableParagraphStyle)

tabStops

– (NSArray *)**tabStops**

Returns the NSTextTab objects, sorted by location, that define the tab stops for the paragraph style.

See also: – **location** (NSTextTab), – **setTabStops:** (NSMutableParagraphStyle),
– **addTabStop:** (NSMutableParagraphStyle), – **removeTabStop:** (NSMutableParagraphStyle)

tailIndent

– (float)**tailIndent**

Returns the distance in points from the margin of a text container to the end of lines. If positive, this is the distance from the leading margin (for example, the left margin in left-to-right text). If zero or negative, it's the distance from the trailing margin.

For example, a paragraph style designed to fit exactly in a 2-inch wide container has a head indent of 0.0 and a tail indent of 0.0. One designed to fit with a quarter-inch margin has a head indent of 0.25 and a tail indent of –0.25.

See also: – **headIndent**, – **firstLineHeadIndent**, – **setTailIndent:** (NSMutableParagraphStyle)