| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/FontMetrics.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/awt/FontFormatException.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/Frame.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/FontMetrics.html)    [**NO FRAMES**](http://docs.google.com/FontMetrics.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#2s8eyo1) | [METHOD](#3rdcrjn) |

## **java.awt**

Class FontMetrics

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **java.awt.FontMetrics**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html)

public abstract class **FontMetrics**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Serializable](http://docs.google.com/java/io/Serializable.html)

The FontMetrics class defines a font metrics object, which encapsulates information about the rendering of a particular font on a particular screen.

**Note to subclassers**: Since many of these methods form closed, mutually recursive loops, you must take care that you implement at least one of the methods in each such loop to prevent infinite recursion when your subclass is used. In particular, the following is the minimal suggested set of methods to override in order to ensure correctness and prevent infinite recursion (though other subsets are equally feasible):

* [getAscent()](http://docs.google.com/java/awt/FontMetrics.html#getAscent())
* [getLeading()](http://docs.google.com/java/awt/FontMetrics.html#getLeading())
* [getMaxAdvance()](http://docs.google.com/java/awt/FontMetrics.html#getMaxAdvance())
* [charWidth(char)](http://docs.google.com/java/awt/FontMetrics.html#charWidth(char))
* [charsWidth(char[], int, int)](http://docs.google.com/java/awt/FontMetrics.html#charsWidth(char%5B%5D,%20int,%20int))

 Note that the implementations of these methods are inefficient, so they are usually overridden with more efficient toolkit-specific implementations.

When an application asks to place a character at the position (*x*, *y*), the character is placed so that its reference point (shown as the dot in the accompanying image) is put at that position. The reference point specifies a horizontal line called the *baseline* of the character. In normal printing, the baselines of characters should align.

In addition, every character in a font has an *ascent*, a *descent*, and an *advance width*. The ascent is the amount by which the character ascends above the baseline. The descent is the amount by which the character descends below the baseline. The advance width indicates the position at which AWT should place the next character.

An array of characters or a string can also have an ascent, a descent, and an advance width. The ascent of the array is the maximum ascent of any character in the array. The descent is the maximum descent of any character in the array. The advance width is the sum of the advance widths of each of the characters in the character array. The advance of a String is the distance along the baseline of the String. This distance is the width that should be used for centering or right-aligning the String.

Note that the advance of a String is not necessarily the sum of the advances of its characters measured in isolation because the width of a character can vary depending on its context. For example, in Arabic text, the shape of a character can change in order to connect to other characters. Also, in some scripts, certain character sequences can be represented by a single shape, called a *ligature*. Measuring characters individually does not account for these transformations.

Font metrics are baseline-relative, meaning that they are generally independent of the rotation applied to the font (modulo possible grid hinting effects). See [Font](http://docs.google.com/java/awt/Font.html).

**Since:** JDK1.0 **See Also:**[Font](http://docs.google.com/java/awt/Font.html), [Serialized Form](http://docs.google.com/serialized-form.html#java.awt.FontMetrics)

| **Field Summary** | |
| --- | --- |
| protected  [Font](http://docs.google.com/java/awt/Font.html) | [**font**](http://docs.google.com/java/awt/FontMetrics.html#font)            The actual [Font](http://docs.google.com/java/awt/Font.html) from which the font metrics are created. |

| **Constructor Summary** | |
| --- | --- |
| protected | [**FontMetrics**](http://docs.google.com/java/awt/FontMetrics.html#FontMetrics(java.awt.Font))([Font](http://docs.google.com/java/awt/Font.html) font)            Creates a new FontMetrics object for finding out height and width information about the specified Font and specific character glyphs in that Font. |

| **Method Summary** | |
| --- | --- |
| int | [**bytesWidth**](http://docs.google.com/java/awt/FontMetrics.html#bytesWidth(byte%5B%5D,%20int,%20int))(byte[] data, int off, int len)            Returns the total advance width for showing the specified array of bytes in this Font. |
| int | [**charsWidth**](http://docs.google.com/java/awt/FontMetrics.html#charsWidth(char%5B%5D,%20int,%20int))(char[] data, int off, int len)            Returns the total advance width for showing the specified array of characters in this Font. |
| int | [**charWidth**](http://docs.google.com/java/awt/FontMetrics.html#charWidth(char))(char ch)            Returns the advance width of the specified character in this Font. |
| int | [**charWidth**](http://docs.google.com/java/awt/FontMetrics.html#charWidth(int))(int codePoint)            Returns the advance width of the specified character in this Font. |
| int | [**getAscent**](http://docs.google.com/java/awt/FontMetrics.html#getAscent())()            Determines the *font ascent* of the Font described by this FontMetrics object. |
| int | [**getDescent**](http://docs.google.com/java/awt/FontMetrics.html#getDescent())()            Determines the *font descent* of the Font described by this FontMetrics object. |
| [Font](http://docs.google.com/java/awt/Font.html) | [**getFont**](http://docs.google.com/java/awt/FontMetrics.html#getFont())()            Gets the Font described by this FontMetrics object. |
| [FontRenderContext](http://docs.google.com/java/awt/font/FontRenderContext.html) | [**getFontRenderContext**](http://docs.google.com/java/awt/FontMetrics.html#getFontRenderContext())()            Gets the FontRenderContext used by this FontMetrics object to measure text. |
| int | [**getHeight**](http://docs.google.com/java/awt/FontMetrics.html#getHeight())()            Gets the standard height of a line of text in this font. |
| int | [**getLeading**](http://docs.google.com/java/awt/FontMetrics.html#getLeading())()            Determines the *standard leading* of the Font described by this FontMetrics object. |
| [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) | [**getLineMetrics**](http://docs.google.com/java/awt/FontMetrics.html#getLineMetrics(char%5B%5D,%20int,%20int,%20java.awt.Graphics))(char[] chars, int beginIndex, int limit, [Graphics](http://docs.google.com/java/awt/Graphics.html) context)            Returns the [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) object for the specified character array in the specified [Graphics](http://docs.google.com/java/awt/Graphics.html) context. |
| [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) | [**getLineMetrics**](http://docs.google.com/java/awt/FontMetrics.html#getLineMetrics(java.text.CharacterIterator,%20int,%20int,%20java.awt.Graphics))([CharacterIterator](http://docs.google.com/java/text/CharacterIterator.html) ci, int beginIndex, int limit, [Graphics](http://docs.google.com/java/awt/Graphics.html) context)            Returns the [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) object for the specified [CharacterIterator](http://docs.google.com/java/text/CharacterIterator.html) in the specified [Graphics](http://docs.google.com/java/awt/Graphics.html) context. |
| [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) | [**getLineMetrics**](http://docs.google.com/java/awt/FontMetrics.html#getLineMetrics(java.lang.String,%20java.awt.Graphics))([String](http://docs.google.com/java/lang/String.html) str, [Graphics](http://docs.google.com/java/awt/Graphics.html) context)            Returns the [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) object for the specified String in the specified [Graphics](http://docs.google.com/java/awt/Graphics.html) context. |
| [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) | [**getLineMetrics**](http://docs.google.com/java/awt/FontMetrics.html#getLineMetrics(java.lang.String,%20int,%20int,%20java.awt.Graphics))([String](http://docs.google.com/java/lang/String.html) str, int beginIndex, int limit, [Graphics](http://docs.google.com/java/awt/Graphics.html) context)            Returns the [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) object for the specified String in the specified [Graphics](http://docs.google.com/java/awt/Graphics.html) context. |
| int | [**getMaxAdvance**](http://docs.google.com/java/awt/FontMetrics.html#getMaxAdvance())()            Gets the maximum advance width of any character in this Font. |
| int | [**getMaxAscent**](http://docs.google.com/java/awt/FontMetrics.html#getMaxAscent())()            Determines the maximum ascent of the Font described by this FontMetrics object. |
| [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) | [**getMaxCharBounds**](http://docs.google.com/java/awt/FontMetrics.html#getMaxCharBounds(java.awt.Graphics))([Graphics](http://docs.google.com/java/awt/Graphics.html) context)            Returns the bounds for the character with the maximum bounds in the specified Graphics context. |
| int | [**getMaxDecent**](http://docs.google.com/java/awt/FontMetrics.html#getMaxDecent())()  **Deprecated.** *As of JDK version 1.1.1, replaced by getMaxDescent().* |
| int | [**getMaxDescent**](http://docs.google.com/java/awt/FontMetrics.html#getMaxDescent())()            Determines the maximum descent of the Font described by this FontMetrics object. |
| [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) | [**getStringBounds**](http://docs.google.com/java/awt/FontMetrics.html#getStringBounds(char%5B%5D,%20int,%20int,%20java.awt.Graphics))(char[] chars, int beginIndex, int limit, [Graphics](http://docs.google.com/java/awt/Graphics.html) context)            Returns the bounds of the specified array of characters in the specified Graphics context. |
| [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) | [**getStringBounds**](http://docs.google.com/java/awt/FontMetrics.html#getStringBounds(java.text.CharacterIterator,%20int,%20int,%20java.awt.Graphics))([CharacterIterator](http://docs.google.com/java/text/CharacterIterator.html) ci, int beginIndex, int limit, [Graphics](http://docs.google.com/java/awt/Graphics.html) context)            Returns the bounds of the characters indexed in the specified CharacterIterator in the specified Graphics context. |
| [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) | [**getStringBounds**](http://docs.google.com/java/awt/FontMetrics.html#getStringBounds(java.lang.String,%20java.awt.Graphics))([String](http://docs.google.com/java/lang/String.html) str, [Graphics](http://docs.google.com/java/awt/Graphics.html) context)            Returns the bounds of the specified String in the specified Graphics context. |
| [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) | [**getStringBounds**](http://docs.google.com/java/awt/FontMetrics.html#getStringBounds(java.lang.String,%20int,%20int,%20java.awt.Graphics))([String](http://docs.google.com/java/lang/String.html) str, int beginIndex, int limit, [Graphics](http://docs.google.com/java/awt/Graphics.html) context)            Returns the bounds of the specified String in the specified Graphics context. |
| int[] | [**getWidths**](http://docs.google.com/java/awt/FontMetrics.html#getWidths())()            Gets the advance widths of the first 256 characters in the Font. |
| boolean | [**hasUniformLineMetrics**](http://docs.google.com/java/awt/FontMetrics.html#hasUniformLineMetrics())()            Checks to see if the Font has uniform line metrics. |
| int | [**stringWidth**](http://docs.google.com/java/awt/FontMetrics.html#stringWidth(java.lang.String))([String](http://docs.google.com/java/lang/String.html) str)            Returns the total advance width for showing the specified String in this Font. |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/java/awt/FontMetrics.html#toString())()            Returns a representation of this FontMetrics object's values as a String. |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [clone](http://docs.google.com/java/lang/Object.html#clone()), [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

| **Field Detail** |
| --- |

### font

protected [Font](http://docs.google.com/java/awt/Font.html) **font**

The actual [Font](http://docs.google.com/java/awt/Font.html) from which the font metrics are created. This cannot be null.

**See Also:**[getFont()](http://docs.google.com/java/awt/FontMetrics.html#getFont())

| **Constructor Detail** |
| --- |

### FontMetrics

protected **FontMetrics**([Font](http://docs.google.com/java/awt/Font.html) font)

Creates a new FontMetrics object for finding out height and width information about the specified Font and specific character glyphs in that Font.

**Parameters:**font - the Font**See Also:**[Font](http://docs.google.com/java/awt/Font.html)

| **Method Detail** |
| --- |

### getFont

public [Font](http://docs.google.com/java/awt/Font.html) **getFont**()

Gets the Font described by this FontMetrics object.

**Returns:**the Font described by this FontMetrics object.

### getFontRenderContext

public [FontRenderContext](http://docs.google.com/java/awt/font/FontRenderContext.html) **getFontRenderContext**()

Gets the FontRenderContext used by this FontMetrics object to measure text.

Note that methods in this class which take a Graphics parameter measure text using the FontRenderContext of that Graphics object, and not this FontRenderContext

**Returns:**the FontRenderContext used by this FontMetrics object.**Since:** 1.6

### getLeading

public int **getLeading**()

Determines the *standard leading* of the Font described by this FontMetrics object. The standard leading, or interline spacing, is the logical amount of space to be reserved between the descent of one line of text and the ascent of the next line. The height metric is calculated to include this extra space.

**Returns:**the standard leading of the Font.**See Also:**[getHeight()](http://docs.google.com/java/awt/FontMetrics.html#getHeight()), [getAscent()](http://docs.google.com/java/awt/FontMetrics.html#getAscent()), [getDescent()](http://docs.google.com/java/awt/FontMetrics.html#getDescent())

### getAscent

public int **getAscent**()

Determines the *font ascent* of the Font described by this FontMetrics object. The font ascent is the distance from the font's baseline to the top of most alphanumeric characters. Some characters in the Font might extend above the font ascent line.

**Returns:**the font ascent of the Font.**See Also:**[getMaxAscent()](http://docs.google.com/java/awt/FontMetrics.html#getMaxAscent())

### getDescent

public int **getDescent**()

Determines the *font descent* of the Font described by this FontMetrics object. The font descent is the distance from the font's baseline to the bottom of most alphanumeric characters with descenders. Some characters in the Font might extend below the font descent line.

**Returns:**the font descent of the Font.**See Also:**[getMaxDescent()](http://docs.google.com/java/awt/FontMetrics.html#getMaxDescent())

### getHeight

public int **getHeight**()

Gets the standard height of a line of text in this font. This is the distance between the baseline of adjacent lines of text. It is the sum of the leading + ascent + descent. Due to rounding this may not be the same as getAscent() + getDescent() + getLeading(). There is no guarantee that lines of text spaced at this distance are disjoint; such lines may overlap if some characters overshoot either the standard ascent or the standard descent metric.

**Returns:**the standard height of the font.**See Also:**[getLeading()](http://docs.google.com/java/awt/FontMetrics.html#getLeading()), [getAscent()](http://docs.google.com/java/awt/FontMetrics.html#getAscent()), [getDescent()](http://docs.google.com/java/awt/FontMetrics.html#getDescent())

### getMaxAscent

public int **getMaxAscent**()

Determines the maximum ascent of the Font described by this FontMetrics object. No character extends further above the font's baseline than this height.

**Returns:**the maximum ascent of any character in the Font.**See Also:**[getAscent()](http://docs.google.com/java/awt/FontMetrics.html#getAscent())

### getMaxDescent

public int **getMaxDescent**()

Determines the maximum descent of the Font described by this FontMetrics object. No character extends further below the font's baseline than this height.

**Returns:**the maximum descent of any character in the Font.**See Also:**[getDescent()](http://docs.google.com/java/awt/FontMetrics.html#getDescent())

### getMaxDecent

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
public int **getMaxDecent**()

**Deprecated.** *As of JDK version 1.1.1, replaced by getMaxDescent().*

For backward compatibility only.

**Returns:**the maximum descent of any character in the Font.**See Also:**[getMaxDescent()](http://docs.google.com/java/awt/FontMetrics.html#getMaxDescent())

### getMaxAdvance

public int **getMaxAdvance**()

Gets the maximum advance width of any character in this Font. The advance is the distance from the leftmost point to the rightmost point on the string's baseline. The advance of a String is not necessarily the sum of the advances of its characters.

**Returns:**the maximum advance width of any character in the Font, or -1 if the maximum advance width is not known.

### charWidth

public int **charWidth**(int codePoint)

Returns the advance width of the specified character in this Font. The advance is the distance from the leftmost point to the rightmost point on the character's baseline. Note that the advance of a String is not necessarily the sum of the advances of its characters.

This method doesn't validate the specified character to be a valid Unicode code point. The caller must validate the character value using [Character.isValidCodePoint](http://docs.google.com/java/lang/Character.html#isValidCodePoint(int)) if necessary.

**Parameters:**codePoint - the character (Unicode code point) to be measured **Returns:**the advance width of the specified character in the Font described by this FontMetrics object.**See Also:**[charsWidth(char[], int, int)](http://docs.google.com/java/awt/FontMetrics.html#charsWidth(char%5B%5D,%20int,%20int)), [stringWidth(String)](http://docs.google.com/java/awt/FontMetrics.html#stringWidth(java.lang.String))

### charWidth

public int **charWidth**(char ch)

Returns the advance width of the specified character in this Font. The advance is the distance from the leftmost point to the rightmost point on the character's baseline. Note that the advance of a String is not necessarily the sum of the advances of its characters.

**Note:** This method cannot handle  [supplementary characters](http://docs.google.com/lang/Character.html#supplementary). To support all Unicode characters, including supplementary characters, use the [charWidth(int)](http://docs.google.com/java/awt/FontMetrics.html#charWidth(int)) method.

**Parameters:**ch - the character to be measured **Returns:**the advance width of the specified character in the Font described by this FontMetrics object.**See Also:**[charsWidth(char[], int, int)](http://docs.google.com/java/awt/FontMetrics.html#charsWidth(char%5B%5D,%20int,%20int)), [stringWidth(String)](http://docs.google.com/java/awt/FontMetrics.html#stringWidth(java.lang.String))

### stringWidth

public int **stringWidth**([String](http://docs.google.com/java/lang/String.html) str)

Returns the total advance width for showing the specified String in this Font. The advance is the distance from the leftmost point to the rightmost point on the string's baseline.

Note that the advance of a String is not necessarily the sum of the advances of its characters.

**Parameters:**str - the String to be measured **Returns:**the advance width of the specified String in the Font described by this FontMetrics. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if str is null.**See Also:**[bytesWidth(byte[], int, int)](http://docs.google.com/java/awt/FontMetrics.html#bytesWidth(byte%5B%5D,%20int,%20int)), [charsWidth(char[], int, int)](http://docs.google.com/java/awt/FontMetrics.html#charsWidth(char%5B%5D,%20int,%20int)), [getStringBounds(String, Graphics)](http://docs.google.com/java/awt/FontMetrics.html#getStringBounds(java.lang.String,%20java.awt.Graphics))

### charsWidth

public int **charsWidth**(char[] data,  
 int off,  
 int len)

Returns the total advance width for showing the specified array of characters in this Font. The advance is the distance from the leftmost point to the rightmost point on the string's baseline. The advance of a String is not necessarily the sum of the advances of its characters. This is equivalent to measuring a String of the characters in the specified range.

**Parameters:**data - the array of characters to be measuredoff - the start offset of the characters in the arraylen - the number of characters to be measured from the array **Returns:**the advance width of the subarray of the specified char array in the font described by this FontMetrics object. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if data is null. [IndexOutOfBoundsException](http://docs.google.com/java/lang/IndexOutOfBoundsException.html) - if the off and len arguments index characters outside the bounds of the data array.**See Also:**[charWidth(int)](http://docs.google.com/java/awt/FontMetrics.html#charWidth(int)), [charWidth(char)](http://docs.google.com/java/awt/FontMetrics.html#charWidth(char)), [bytesWidth(byte[], int, int)](http://docs.google.com/java/awt/FontMetrics.html#bytesWidth(byte%5B%5D,%20int,%20int)), [stringWidth(String)](http://docs.google.com/java/awt/FontMetrics.html#stringWidth(java.lang.String))

### bytesWidth

public int **bytesWidth**(byte[] data,  
 int off,  
 int len)

Returns the total advance width for showing the specified array of bytes in this Font. The advance is the distance from the leftmost point to the rightmost point on the string's baseline. The advance of a String is not necessarily the sum of the advances of its characters. This is equivalent to measuring a String of the characters in the specified range.

**Parameters:**data - the array of bytes to be measuredoff - the start offset of the bytes in the arraylen - the number of bytes to be measured from the array **Returns:**the advance width of the subarray of the specified byte array in the Font described by this FontMetrics object. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if data is null. [IndexOutOfBoundsException](http://docs.google.com/java/lang/IndexOutOfBoundsException.html) - if the off and len arguments index bytes outside the bounds of the data array.**See Also:**[charsWidth(char[], int, int)](http://docs.google.com/java/awt/FontMetrics.html#charsWidth(char%5B%5D,%20int,%20int)), [stringWidth(String)](http://docs.google.com/java/awt/FontMetrics.html#stringWidth(java.lang.String))

### getWidths

public int[] **getWidths**()

Gets the advance widths of the first 256 characters in the Font. The advance is the distance from the leftmost point to the rightmost point on the character's baseline. Note that the advance of a String is not necessarily the sum of the advances of its characters.

**Returns:**an array storing the advance widths of the characters in the Font described by this FontMetrics object.

### hasUniformLineMetrics

public boolean **hasUniformLineMetrics**()

Checks to see if the Font has uniform line metrics. A composite font may consist of several different fonts to cover various character sets. In such cases, the FontLineMetrics objects are not uniform. Different fonts may have a different ascent, descent, metrics and so on. This information is sometimes necessary for line measuring and line breaking.

**Returns:**true if the font has uniform line metrics; false otherwise.**See Also:**[Font.hasUniformLineMetrics()](http://docs.google.com/java/awt/Font.html#hasUniformLineMetrics())

### getLineMetrics

public [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) **getLineMetrics**([String](http://docs.google.com/java/lang/String.html) str,  
 [Graphics](http://docs.google.com/java/awt/Graphics.html) context)

Returns the [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) object for the specified String in the specified [Graphics](http://docs.google.com/java/awt/Graphics.html) context.

**Parameters:**str - the specified Stringcontext - the specified Graphics context **Returns:**a LineMetrics object created with the specified String and Graphics context.**See Also:**[Font.getLineMetrics(String, FontRenderContext)](http://docs.google.com/java/awt/Font.html#getLineMetrics(java.lang.String,%20java.awt.font.FontRenderContext))

### getLineMetrics

public [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) **getLineMetrics**([String](http://docs.google.com/java/lang/String.html) str,  
 int beginIndex,  
 int limit,  
 [Graphics](http://docs.google.com/java/awt/Graphics.html) context)

Returns the [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) object for the specified String in the specified [Graphics](http://docs.google.com/java/awt/Graphics.html) context.

**Parameters:**str - the specified StringbeginIndex - the initial offset of strlimit - the end offset of strcontext - the specified Graphics context **Returns:**a LineMetrics object created with the specified String and Graphics context.**See Also:**[Font.getLineMetrics(String, int, int, FontRenderContext)](http://docs.google.com/java/awt/Font.html#getLineMetrics(java.lang.String,%20int,%20int,%20java.awt.font.FontRenderContext))

### getLineMetrics

public [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) **getLineMetrics**(char[] chars,  
 int beginIndex,  
 int limit,  
 [Graphics](http://docs.google.com/java/awt/Graphics.html) context)

Returns the [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) object for the specified character array in the specified [Graphics](http://docs.google.com/java/awt/Graphics.html) context.

**Parameters:**chars - the specified character arraybeginIndex - the initial offset of charslimit - the end offset of charscontext - the specified Graphics context **Returns:**a LineMetrics object created with the specified character array and Graphics context.**See Also:**[Font.getLineMetrics(char[], int, int, FontRenderContext)](http://docs.google.com/java/awt/Font.html#getLineMetrics(char%5B%5D,%20int,%20int,%20java.awt.font.FontRenderContext))

### getLineMetrics

public [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) **getLineMetrics**([CharacterIterator](http://docs.google.com/java/text/CharacterIterator.html) ci,  
 int beginIndex,  
 int limit,  
 [Graphics](http://docs.google.com/java/awt/Graphics.html) context)

Returns the [LineMetrics](http://docs.google.com/java/awt/font/LineMetrics.html) object for the specified [CharacterIterator](http://docs.google.com/java/text/CharacterIterator.html) in the specified [Graphics](http://docs.google.com/java/awt/Graphics.html) context.

**Parameters:**ci - the specified CharacterIteratorbeginIndex - the initial offset in cilimit - the end index of cicontext - the specified Graphics context **Returns:**a LineMetrics object created with the specified arguments.**See Also:**[Font.getLineMetrics(CharacterIterator, int, int, FontRenderContext)](http://docs.google.com/java/awt/Font.html#getLineMetrics(java.text.CharacterIterator,%20int,%20int,%20java.awt.font.FontRenderContext))

### getStringBounds

public [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) **getStringBounds**([String](http://docs.google.com/java/lang/String.html) str,  
 [Graphics](http://docs.google.com/java/awt/Graphics.html) context)

Returns the bounds of the specified String in the specified Graphics context. The bounds is used to layout the String.

Note: The returned bounds is in baseline-relative coordinates (see [class notes](http://docs.google.com/java/awt/FontMetrics.html)).

**Parameters:**str - the specified Stringcontext - the specified Graphics context **Returns:**a [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) that is the bounding box of the specified String in the specified Graphics context.**See Also:**[Font.getStringBounds(String, FontRenderContext)](http://docs.google.com/java/awt/Font.html#getStringBounds(java.lang.String,%20java.awt.font.FontRenderContext))

### getStringBounds

public [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) **getStringBounds**([String](http://docs.google.com/java/lang/String.html) str,  
 int beginIndex,  
 int limit,  
 [Graphics](http://docs.google.com/java/awt/Graphics.html) context)

Returns the bounds of the specified String in the specified Graphics context. The bounds is used to layout the String.

Note: The returned bounds is in baseline-relative coordinates (see [class notes](http://docs.google.com/java/awt/FontMetrics.html)).

**Parameters:**str - the specified StringbeginIndex - the offset of the beginning of strlimit - the end offset of strcontext - the specified Graphics context **Returns:**a Rectangle2D that is the bounding box of the specified String in the specified Graphics context.**See Also:**[Font.getStringBounds(String, int, int, FontRenderContext)](http://docs.google.com/java/awt/Font.html#getStringBounds(java.lang.String,%20int,%20int,%20java.awt.font.FontRenderContext))

### getStringBounds

public [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) **getStringBounds**(char[] chars,  
 int beginIndex,  
 int limit,  
 [Graphics](http://docs.google.com/java/awt/Graphics.html) context)

Returns the bounds of the specified array of characters in the specified Graphics context. The bounds is used to layout the String created with the specified array of characters, beginIndex and limit.

Note: The returned bounds is in baseline-relative coordinates (see [class notes](http://docs.google.com/java/awt/FontMetrics.html)).

**Parameters:**chars - an array of charactersbeginIndex - the initial offset of the array of characterslimit - the end offset of the array of characterscontext - the specified Graphics context **Returns:**a Rectangle2D that is the bounding box of the specified character array in the specified Graphics context.**See Also:**[Font.getStringBounds(char[], int, int, FontRenderContext)](http://docs.google.com/java/awt/Font.html#getStringBounds(char%5B%5D,%20int,%20int,%20java.awt.font.FontRenderContext))

### getStringBounds

public [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) **getStringBounds**([CharacterIterator](http://docs.google.com/java/text/CharacterIterator.html) ci,  
 int beginIndex,  
 int limit,  
 [Graphics](http://docs.google.com/java/awt/Graphics.html) context)

Returns the bounds of the characters indexed in the specified CharacterIterator in the specified Graphics context.

Note: The returned bounds is in baseline-relative coordinates (see [class notes](http://docs.google.com/java/awt/FontMetrics.html)).

**Parameters:**ci - the specified CharacterIteratorbeginIndex - the initial offset in cilimit - the end index of cicontext - the specified Graphics context **Returns:**a Rectangle2D that is the bounding box of the characters indexed in the specified CharacterIterator in the specified Graphics context.**See Also:**[Font.getStringBounds(CharacterIterator, int, int, FontRenderContext)](http://docs.google.com/java/awt/Font.html#getStringBounds(java.text.CharacterIterator,%20int,%20int,%20java.awt.font.FontRenderContext))

### getMaxCharBounds

public [Rectangle2D](http://docs.google.com/java/awt/geom/Rectangle2D.html) **getMaxCharBounds**([Graphics](http://docs.google.com/java/awt/Graphics.html) context)

Returns the bounds for the character with the maximum bounds in the specified Graphics context.

**Parameters:**context - the specified Graphics context **Returns:**a Rectangle2D that is the bounding box for the character with the maximum bounds.**See Also:**[Font.getMaxCharBounds(FontRenderContext)](http://docs.google.com/java/awt/Font.html#getMaxCharBounds(java.awt.font.FontRenderContext))

### toString

public [String](http://docs.google.com/java/lang/String.html) **toString**()

Returns a representation of this FontMetrics object's values as a String.

**Overrides:**[toString](http://docs.google.com/java/lang/Object.html#toString()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a String representation of this FontMetrics object.**Since:** JDK1.0.

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/FontMetrics.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/java/awt/FontFormatException.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/Frame.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/FontMetrics.html)    [**NO FRAMES**](http://docs.google.com/FontMetrics.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#2s8eyo1) | [METHOD](#3rdcrjn) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).