| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DefaultHighlighter.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/javax/swing/text/DefaultFormatterFactory.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/DefaultHighlighter.DefaultHighlightPainter.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/DefaultHighlighter.html)    [**NO FRAMES**](http://docs.google.com/DefaultHighlighter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#3dy6vkm) | [CONSTR](#1t3h5sf) | [METHOD](#4d34og8) | DETAIL: [FIELD](#17dp8vu) | [CONSTR](#26in1rg) | [METHOD](#35nkun2) |

## **javax.swing.text**

Class DefaultHighlighter

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.swing.text.LayeredHighlighter](http://docs.google.com/javax/swing/text/LayeredHighlighter.html)  
 **javax.swing.text.DefaultHighlighter**

**All Implemented Interfaces:** [Highlighter](http://docs.google.com/javax/swing/text/Highlighter.html) **Direct Known Subclasses:** [BasicTextUI.BasicHighlighter](http://docs.google.com/javax/swing/plaf/basic/BasicTextUI.BasicHighlighter.html)

public class **DefaultHighlighter**extends [LayeredHighlighter](http://docs.google.com/javax/swing/text/LayeredHighlighter.html)

Implements the Highlighter interfaces. Implements a simple highlight painter that renders in a solid color.

**See Also:**[Highlighter](http://docs.google.com/javax/swing/text/Highlighter.html)

| **Nested Class Summary** | |
| --- | --- |
| static class | [**DefaultHighlighter.DefaultHighlightPainter**](http://docs.google.com/javax/swing/text/DefaultHighlighter.DefaultHighlightPainter.html)            Simple highlight painter that fills a highlighted area with a solid color. |

| **Nested classes/interfaces inherited from class javax.swing.text.**[**LayeredHighlighter**](http://docs.google.com/javax/swing/text/LayeredHighlighter.html) |
| --- |
| [LayeredHighlighter.LayerPainter](http://docs.google.com/javax/swing/text/LayeredHighlighter.LayerPainter.html) |

| **Nested classes/interfaces inherited from interface javax.swing.text.**[**Highlighter**](http://docs.google.com/javax/swing/text/Highlighter.html) |
| --- |
| [Highlighter.Highlight](http://docs.google.com/javax/swing/text/Highlighter.Highlight.html), [Highlighter.HighlightPainter](http://docs.google.com/javax/swing/text/Highlighter.HighlightPainter.html) |

| **Field Summary** | |
| --- | --- |
| static [LayeredHighlighter.LayerPainter](http://docs.google.com/javax/swing/text/LayeredHighlighter.LayerPainter.html) | [**DefaultPainter**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#DefaultPainter)            Default implementation of LayeredHighlighter.LayerPainter that can be used for painting highlights. |

| **Constructor Summary** | |
| --- | --- |
| [**DefaultHighlighter**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#DefaultHighlighter())()            Creates a new DefaultHighlighther object. |

| **Method Summary** | |
| --- | --- |
| [Object](http://docs.google.com/java/lang/Object.html) | [**addHighlight**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#addHighlight(int,%20int,%20javax.swing.text.Highlighter.HighlightPainter))(int p0, int p1, [Highlighter.HighlightPainter](http://docs.google.com/javax/swing/text/Highlighter.HighlightPainter.html) p)            Adds a highlight to the view. |
| void | [**changeHighlight**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#changeHighlight(java.lang.Object,%20int,%20int))([Object](http://docs.google.com/java/lang/Object.html) tag, int p0, int p1)            Changes a highlight. |
| void | [**deinstall**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#deinstall(javax.swing.text.JTextComponent))([JTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.html) c)            Called when the UI is being removed from the interface of a JTextComponent. |
| boolean | [**getDrawsLayeredHighlights**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#getDrawsLayeredHighlights())() |
| [Highlighter.Highlight](http://docs.google.com/javax/swing/text/Highlighter.Highlight.html)[] | [**getHighlights**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#getHighlights())()            Makes a copy of the highlights. |
| void | [**install**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#install(javax.swing.text.JTextComponent))([JTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.html) c)            Called when the UI is being installed into the interface of a JTextComponent. |
| void | [**paint**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#paint(java.awt.Graphics))([Graphics](http://docs.google.com/java/awt/Graphics.html) g)            Renders the highlights. |
| void | [**paintLayeredHighlights**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#paintLayeredHighlights(java.awt.Graphics,%20int,%20int,%20java.awt.Shape,%20javax.swing.text.JTextComponent,%20javax.swing.text.View))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, int p0, int p1, [Shape](http://docs.google.com/java/awt/Shape.html) viewBounds, [JTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.html) editor, [View](http://docs.google.com/javax/swing/text/View.html) view)            When leaf Views (such as LabelView) are rendering they should call into this method. |
| void | [**removeAllHighlights**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#removeAllHighlights())()            Removes all highlights. |
| void | [**removeHighlight**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#removeHighlight(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) tag)            Removes a highlight from the view. |
| void | [**setDrawsLayeredHighlights**](http://docs.google.com/javax/swing/text/DefaultHighlighter.html#setDrawsLayeredHighlights(boolean))(boolean newValue)            If true, highlights are drawn as the Views draw the text. |

| **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()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [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** |
| --- |

### DefaultPainter

public static final [LayeredHighlighter.LayerPainter](http://docs.google.com/javax/swing/text/LayeredHighlighter.LayerPainter.html) **DefaultPainter**

Default implementation of LayeredHighlighter.LayerPainter that can be used for painting highlights.

As of 1.4 this field is final.

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

### DefaultHighlighter

public **DefaultHighlighter**()

Creates a new DefaultHighlighther object.

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

### paint

public void **paint**([Graphics](http://docs.google.com/java/awt/Graphics.html) g)

Renders the highlights.

**Parameters:**g - the graphics context

### install

public void **install**([JTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.html) c)

Called when the UI is being installed into the interface of a JTextComponent. Installs the editor, and removes any existing highlights.

**Parameters:**c - the editor component**See Also:**[Highlighter.install(javax.swing.text.JTextComponent)](http://docs.google.com/javax/swing/text/Highlighter.html#install(javax.swing.text.JTextComponent))

### deinstall

public void **deinstall**([JTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.html) c)

Called when the UI is being removed from the interface of a JTextComponent.

**Parameters:**c - the component**See Also:**[Highlighter.deinstall(javax.swing.text.JTextComponent)](http://docs.google.com/javax/swing/text/Highlighter.html#deinstall(javax.swing.text.JTextComponent))

### addHighlight

public [Object](http://docs.google.com/java/lang/Object.html) **addHighlight**(int p0,  
 int p1,  
 [Highlighter.HighlightPainter](http://docs.google.com/javax/swing/text/Highlighter.HighlightPainter.html) p)  
 throws [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html)

Adds a highlight to the view. Returns a tag that can be used to refer to the highlight.

**Parameters:**p0 - the start offset of the range to highlight >= 0p1 - the end offset of the range to highlight >= p0p - the painter to use to actually render the highlight **Returns:**an object that can be used as a tag to refer to the highlight **Throws:** [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html) - if the specified location is invalid

### removeHighlight

public void **removeHighlight**([Object](http://docs.google.com/java/lang/Object.html) tag)

Removes a highlight from the view.

**Parameters:**tag - the reference to the highlight

### removeAllHighlights

public void **removeAllHighlights**()

Removes all highlights.

### changeHighlight

public void **changeHighlight**([Object](http://docs.google.com/java/lang/Object.html) tag,  
 int p0,  
 int p1)  
 throws [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html)

Changes a highlight.

**Parameters:**tag - the highlight tagp0 - the beginning of the range >= 0p1 - the end of the range >= p0 **Throws:** [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html) - if the specified location is invalid

### getHighlights

public [Highlighter.Highlight](http://docs.google.com/javax/swing/text/Highlighter.Highlight.html)[] **getHighlights**()

Makes a copy of the highlights. Does not actually clone each highlight, but only makes references to them.

**Returns:**the copy**See Also:**[Highlighter.getHighlights()](http://docs.google.com/javax/swing/text/Highlighter.html#getHighlights())

### paintLayeredHighlights

public void **paintLayeredHighlights**([Graphics](http://docs.google.com/java/awt/Graphics.html) g,  
 int p0,  
 int p1,  
 [Shape](http://docs.google.com/java/awt/Shape.html) viewBounds,  
 [JTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.html) editor,  
 [View](http://docs.google.com/javax/swing/text/View.html) view)

When leaf Views (such as LabelView) are rendering they should call into this method. If a highlight is in the given region it will be drawn immediately.

**Specified by:**[paintLayeredHighlights](http://docs.google.com/javax/swing/text/LayeredHighlighter.html#paintLayeredHighlights(java.awt.Graphics,%20int,%20int,%20java.awt.Shape,%20javax.swing.text.JTextComponent,%20javax.swing.text.View)) in class [LayeredHighlighter](http://docs.google.com/javax/swing/text/LayeredHighlighter.html) **Parameters:**g - Graphics used to drawp0 - starting offset of viewp1 - ending offset of viewviewBounds - Bounds of Vieweditor - JTextComponentview - View instance being rendered

### setDrawsLayeredHighlights

public void **setDrawsLayeredHighlights**(boolean newValue)

If true, highlights are drawn as the Views draw the text. That is the Views will call into paintLayeredHighlight which will result in a rectangle being drawn before the text is drawn (if the offsets are in a highlighted region that is). For this to work the painter supplied must be an instance of LayeredHighlightPainter.

### getDrawsLayeredHighlights

public boolean **getDrawsLayeredHighlights**()

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DefaultHighlighter.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/javax/swing/text/DefaultFormatterFactory.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/DefaultHighlighter.DefaultHighlightPainter.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/DefaultHighlighter.html)    [**NO FRAMES**](http://docs.google.com/DefaultHighlighter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#3dy6vkm) | [CONSTR](#1t3h5sf) | [METHOD](#4d34og8) | DETAIL: [FIELD](#17dp8vu) | [CONSTR](#26in1rg) | [METHOD](#35nkun2) |

[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).