| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/GlyphView.GlyphPainter.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/GlyphView.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/Highlighter.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/GlyphView.GlyphPainter.html)    [**NO FRAMES**](http://docs.google.com/GlyphView.GlyphPainter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

## **javax.swing.text**

Class GlyphView.GlyphPainter

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

**Enclosing class:**[GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html)

public abstract static class **GlyphView.GlyphPainter**extends [Object](http://docs.google.com/java/lang/Object.html)

A class to perform rendering of the glyphs. This can be implemented to be stateless, or to hold some information as a cache to facilitate faster rendering and model/view translation. At a minimum, the GlyphPainter allows a View implementation to perform its duties independant of a particular version of JVM and selection of capabilities (i.e. shaping for i18n, etc).

**Since:** 1.3

| **Constructor Summary** | |
| --- | --- |
| [**GlyphView.GlyphPainter**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#GlyphView.GlyphPainter())() |

| **Method Summary** | |
| --- | --- |
| abstract  float | [**getAscent**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getAscent(javax.swing.text.GlyphView))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v) |
| abstract  int | [**getBoundedPosition**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getBoundedPosition(javax.swing.text.GlyphView,%20int,%20float,%20float))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, int p0, float x, float len)            Determines the model location that represents the maximum advance that fits within the given span. |
| abstract  float | [**getDescent**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getDescent(javax.swing.text.GlyphView))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v) |
| abstract  float | [**getHeight**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getHeight(javax.swing.text.GlyphView))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v) |
| int | [**getNextVisualPositionFrom**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getNextVisualPositionFrom(javax.swing.text.GlyphView,%20int,%20javax.swing.text.Position.Bias,%20java.awt.Shape,%20int,%20javax.swing.text.Position.Bias%5B%5D))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, int pos, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) b, [Shape](http://docs.google.com/java/awt/Shape.html) a, int direction, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html)[] biasRet)            Provides a way to determine the next visually represented model location that one might place a caret. |
| [GlyphView.GlyphPainter](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html) | [**getPainter**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getPainter(javax.swing.text.GlyphView,%20int,%20int))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, int p0, int p1)            Create a painter to use for the given GlyphView. |
| abstract  float | [**getSpan**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#getSpan(javax.swing.text.GlyphView,%20int,%20int,%20javax.swing.text.TabExpander,%20float))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, int p0, int p1, [TabExpander](http://docs.google.com/javax/swing/text/TabExpander.html) e, float x)            Determine the span the glyphs given a start location (for tab expansion). |
| abstract  [Shape](http://docs.google.com/java/awt/Shape.html) | [**modelToView**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#modelToView(javax.swing.text.GlyphView,%20int,%20javax.swing.text.Position.Bias,%20java.awt.Shape))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, int pos, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) bias, [Shape](http://docs.google.com/java/awt/Shape.html) a)            Provides a mapping from the document model coordinate space to the coordinate space of the view mapped to it. |
| abstract  void | [**paint**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#paint(javax.swing.text.GlyphView,%20java.awt.Graphics,%20java.awt.Shape,%20int,%20int))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, [Graphics](http://docs.google.com/java/awt/Graphics.html) g, [Shape](http://docs.google.com/java/awt/Shape.html) a, int p0, int p1)            Paint the glyphs representing the given range. |
| abstract  int | [**viewToModel**](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html#viewToModel(javax.swing.text.GlyphView,%20float,%20float,%20java.awt.Shape,%20javax.swing.text.Position.Bias%5B%5D))([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v, float x, float y, [Shape](http://docs.google.com/java/awt/Shape.html) a, [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html)[] biasReturn)            Provides a mapping from the view coordinate space to the logical coordinate space of the model. |

| **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)) |

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

### GlyphView.GlyphPainter

public **GlyphView.GlyphPainter**()

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

### getSpan

public abstract float **getSpan**([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v,  
 int p0,  
 int p1,  
 [TabExpander](http://docs.google.com/javax/swing/text/TabExpander.html) e,  
 float x)

Determine the span the glyphs given a start location (for tab expansion).

### getHeight

public abstract float **getHeight**([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v)

### getAscent

public abstract float **getAscent**([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v)

### getDescent

public abstract float **getDescent**([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v)

### paint

public abstract void **paint**([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v,  
 [Graphics](http://docs.google.com/java/awt/Graphics.html) g,  
 [Shape](http://docs.google.com/java/awt/Shape.html) a,  
 int p0,  
 int p1)

Paint the glyphs representing the given range.

### modelToView

public abstract [Shape](http://docs.google.com/java/awt/Shape.html) **modelToView**([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v,  
 int pos,  
 [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) bias,  
 [Shape](http://docs.google.com/java/awt/Shape.html) a)  
 throws [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html)

Provides a mapping from the document model coordinate space to the coordinate space of the view mapped to it. This is shared by the broken views.

**Parameters:**v - the GlyphView containing the destination coordinate spacepos - the position to convertbias - either Position.Bias.Forward or Position.Bias.Backwarda - Bounds of the View **Returns:**the bounding box of the given position **Throws:** [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html) - if the given position does not represent a valid location in the associated document**See Also:**[View.modelToView(int, java.awt.Shape, javax.swing.text.Position.Bias)](http://docs.google.com/javax/swing/text/View.html#modelToView(int,%20java.awt.Shape,%20javax.swing.text.Position.Bias))

### viewToModel

public abstract int **viewToModel**([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v,  
 float x,  
 float y,  
 [Shape](http://docs.google.com/java/awt/Shape.html) a,  
 [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html)[] biasReturn)

Provides a mapping from the view coordinate space to the logical coordinate space of the model.

**Parameters:**v - the GlyphView to provide a mapping forx - the X coordinatey - the Y coordinatea - the allocated region to render intobiasReturn - either Position.Bias.Forward or Position.Bias.Backward is returned as the zero-th element of this array **Returns:**the location within the model that best represents the given point of view**See Also:**[View.viewToModel(float, float, java.awt.Shape, javax.swing.text.Position.Bias[])](http://docs.google.com/javax/swing/text/View.html#viewToModel(float,%20float,%20java.awt.Shape,%20javax.swing.text.Position.Bias%5B%5D))

### getBoundedPosition

public abstract int **getBoundedPosition**([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v,  
 int p0,  
 float x,  
 float len)

Determines the model location that represents the maximum advance that fits within the given span. This could be used to break the given view. The result should be a location just shy of the given advance. This differs from viewToModel which returns the closest position which might be proud of the maximum advance.

**Parameters:**v - the view to find the model location to break at.p0 - the location in the model where the fragment should start it's representation >= 0.x - the graphic location along the axis that the broken view would occupy >= 0. This may be useful for things like tab calculations.len - specifies the distance into the view where a potential break is desired >= 0. **Returns:**the maximum model location possible for a break.**See Also:**[View.breakView(int, int, float, float)](http://docs.google.com/javax/swing/text/View.html#breakView(int,%20int,%20float,%20float))

### getPainter

public [GlyphView.GlyphPainter](http://docs.google.com/javax/swing/text/GlyphView.GlyphPainter.html) **getPainter**([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v,  
 int p0,  
 int p1)

Create a painter to use for the given GlyphView. If the painter carries state it can create another painter to represent a new GlyphView that is being created. If the painter doesn't hold any significant state, it can return itself. The default behavior is to return itself.

**Parameters:**v - the GlyphView to provide a painter forp0 - the starting document offset >= 0p1 - the ending document offset >= p0

### getNextVisualPositionFrom

public int **getNextVisualPositionFrom**([GlyphView](http://docs.google.com/javax/swing/text/GlyphView.html) v,  
 int pos,  
 [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html) b,  
 [Shape](http://docs.google.com/java/awt/Shape.html) a,  
 int direction,  
 [Position.Bias](http://docs.google.com/javax/swing/text/Position.Bias.html)[] biasRet)  
 throws [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html)

Provides a way to determine the next visually represented model location that one might place a caret. Some views may not be visible, they might not be in the same order found in the model, or they just might not allow access to some of the locations in the model.

**Parameters:**v - the view to usepos - the position to convert >= 0b - either Position.Bias.Forward or Position.Bias.Backwarda - the allocated region to render intodirection - the direction from the current position that can be thought of as the arrow keys typically found on a keyboard. This may be SwingConstants.WEST, SwingConstants.EAST, SwingConstants.NORTH, or SwingConstants.SOUTH.biasRet - either Position.Bias.Forward or Position.Bias.Backward is returned as the zero-th element of this array **Returns:**the location within the model that best represents the next location visual position. **Throws:** [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html) [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - for an invalid direction

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/GlyphView.GlyphPainter.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/GlyphView.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/Highlighter.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/GlyphView.GlyphPainter.html)    [**NO FRAMES**](http://docs.google.com/GlyphView.GlyphPainter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

[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).