| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BasicScrollPaneUI.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/plaf/basic/BasicScrollBarUI.TrackListener.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.HSBChangeListener.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/basic/BasicScrollPaneUI.html)    [**NO FRAMES**](http://docs.google.com/BasicScrollPaneUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2et92p0) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: [FIELD](#17dp8vu) | [CONSTR](#44sinio) | [METHOD](#z337ya) |

## **javax.swing.plaf.basic**

Class BasicScrollPaneUI

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.swing.plaf.ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html)  
 [javax.swing.plaf.ScrollPaneUI](http://docs.google.com/javax/swing/plaf/ScrollPaneUI.html)  
 **javax.swing.plaf.basic.BasicScrollPaneUI**

**All Implemented Interfaces:** [ScrollPaneConstants](http://docs.google.com/javax/swing/ScrollPaneConstants.html) **Direct Known Subclasses:** [MetalScrollPaneUI](http://docs.google.com/javax/swing/plaf/metal/MetalScrollPaneUI.html)

public class **BasicScrollPaneUI**extends [ScrollPaneUI](http://docs.google.com/javax/swing/plaf/ScrollPaneUI.html)implements [ScrollPaneConstants](http://docs.google.com/javax/swing/ScrollPaneConstants.html)

A default L&F implementation of ScrollPaneUI.

| **Nested Class Summary** | |
| --- | --- |
| class | [**BasicScrollPaneUI.HSBChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.HSBChangeListener.html)            Horizontal scrollbar listener. |
| protected  class | [**BasicScrollPaneUI.MouseWheelHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.MouseWheelHandler.html)            MouseWheelHandler is an inner class which implements the MouseWheelListener interface. |
| class | [**BasicScrollPaneUI.PropertyChangeHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.PropertyChangeHandler.html) |
| class | [**BasicScrollPaneUI.ViewportChangeHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.ViewportChangeHandler.html)            Listener for viewport events. |
| class | [**BasicScrollPaneUI.VSBChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.VSBChangeListener.html)            Vertical scrollbar listener. |

| **Field Summary** | |
| --- | --- |
| protected  [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) | [**hsbChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#hsbChangeListener) |
| protected  [JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) | [**scrollpane**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#scrollpane) |
| protected  [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) | [**spPropertyChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#spPropertyChangeListener) |
| protected  [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) | [**viewportChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#viewportChangeListener) |
| protected  [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) | [**vsbChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#vsbChangeListener) |

| **Fields inherited from interface javax.swing.**[**ScrollPaneConstants**](http://docs.google.com/javax/swing/ScrollPaneConstants.html) |
| --- |
| [COLUMN\_HEADER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#COLUMN_HEADER), [HORIZONTAL\_SCROLLBAR](http://docs.google.com/javax/swing/ScrollPaneConstants.html#HORIZONTAL_SCROLLBAR), [HORIZONTAL\_SCROLLBAR\_ALWAYS](http://docs.google.com/javax/swing/ScrollPaneConstants.html#HORIZONTAL_SCROLLBAR_ALWAYS), [HORIZONTAL\_SCROLLBAR\_AS\_NEEDED](http://docs.google.com/javax/swing/ScrollPaneConstants.html#HORIZONTAL_SCROLLBAR_AS_NEEDED), [HORIZONTAL\_SCROLLBAR\_NEVER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#HORIZONTAL_SCROLLBAR_NEVER), [HORIZONTAL\_SCROLLBAR\_POLICY](http://docs.google.com/javax/swing/ScrollPaneConstants.html#HORIZONTAL_SCROLLBAR_POLICY), [LOWER\_LEADING\_CORNER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#LOWER_LEADING_CORNER), [LOWER\_LEFT\_CORNER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#LOWER_LEFT_CORNER), [LOWER\_RIGHT\_CORNER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#LOWER_RIGHT_CORNER), [LOWER\_TRAILING\_CORNER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#LOWER_TRAILING_CORNER), [ROW\_HEADER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#ROW_HEADER), [UPPER\_LEADING\_CORNER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#UPPER_LEADING_CORNER), [UPPER\_LEFT\_CORNER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#UPPER_LEFT_CORNER), [UPPER\_RIGHT\_CORNER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#UPPER_RIGHT_CORNER), [UPPER\_TRAILING\_CORNER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#UPPER_TRAILING_CORNER), [VERTICAL\_SCROLLBAR](http://docs.google.com/javax/swing/ScrollPaneConstants.html#VERTICAL_SCROLLBAR), [VERTICAL\_SCROLLBAR\_ALWAYS](http://docs.google.com/javax/swing/ScrollPaneConstants.html#VERTICAL_SCROLLBAR_ALWAYS), [VERTICAL\_SCROLLBAR\_AS\_NEEDED](http://docs.google.com/javax/swing/ScrollPaneConstants.html#VERTICAL_SCROLLBAR_AS_NEEDED), [VERTICAL\_SCROLLBAR\_NEVER](http://docs.google.com/javax/swing/ScrollPaneConstants.html#VERTICAL_SCROLLBAR_NEVER), [VERTICAL\_SCROLLBAR\_POLICY](http://docs.google.com/javax/swing/ScrollPaneConstants.html#VERTICAL_SCROLLBAR_POLICY), [VIEWPORT](http://docs.google.com/javax/swing/ScrollPaneConstants.html#VIEWPORT) |

| **Constructor Summary** | |
| --- | --- |
| [**BasicScrollPaneUI**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#BasicScrollPaneUI())() |

| **Method Summary** | |
| --- | --- |
| protected  [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) | [**createHSBChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#createHSBChangeListener())() |
| protected  [MouseWheelListener](http://docs.google.com/java/awt/event/MouseWheelListener.html) | [**createMouseWheelListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#createMouseWheelListener())()            Creates an instance of MouseWheelListener, which is added to the JScrollPane by installUI(). |
| protected  [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) | [**createPropertyChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#createPropertyChangeListener())()            Creates an instance of PropertyChangeListener that's added to the JScrollPane by installUI(). |
| static [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) | [**createUI**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#createUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) x) |
| protected  [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) | [**createViewportChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#createViewportChangeListener())() |
| protected  [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) | [**createVSBChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#createVSBChangeListener())() |
| int | [**getBaseline**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#getBaseline(javax.swing.JComponent,%20int,%20int))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c, int width, int height)            Returns the baseline. |
| [Component.BaselineResizeBehavior](http://docs.google.com/java/awt/Component.BaselineResizeBehavior.html) | [**getBaselineResizeBehavior**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#getBaselineResizeBehavior(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Returns an enum indicating how the baseline of the component changes as the size changes. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getMaximumSize**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#getMaximumSize(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Returns the specified component's maximum size appropriate for the look and feel. |
| protected  void | [**installDefaults**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#installDefaults(javax.swing.JScrollPane))([JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) scrollpane) |
| protected  void | [**installKeyboardActions**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#installKeyboardActions(javax.swing.JScrollPane))([JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) c) |
| protected  void | [**installListeners**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#installListeners(javax.swing.JScrollPane))([JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) c) |
| void | [**installUI**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#installUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) x)            Configures the specified component appropriate for the look and feel. |
| void | [**paint**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#paint(java.awt.Graphics,%20javax.swing.JComponent))([Graphics](http://docs.google.com/java/awt/Graphics.html) g, [JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Paints the specified component appropriate for the look and feel. |
| protected  void | [**syncScrollPaneWithViewport**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#syncScrollPaneWithViewport())() |
| protected  void | [**uninstallDefaults**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#uninstallDefaults(javax.swing.JScrollPane))([JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) c) |
| protected  void | [**uninstallKeyboardActions**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#uninstallKeyboardActions(javax.swing.JScrollPane))([JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) c) |
| protected  void | [**uninstallListeners**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#uninstallListeners(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c) |
| void | [**uninstallUI**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#uninstallUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Reverses configuration which was done on the specified component during installUI. |
| protected  void | [**updateColumnHeader**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#updateColumnHeader(java.beans.PropertyChangeEvent))([PropertyChangeEvent](http://docs.google.com/java/beans/PropertyChangeEvent.html) e) |
| protected  void | [**updateRowHeader**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#updateRowHeader(java.beans.PropertyChangeEvent))([PropertyChangeEvent](http://docs.google.com/java/beans/PropertyChangeEvent.html) e) |
| protected  void | [**updateScrollBarDisplayPolicy**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#updateScrollBarDisplayPolicy(java.beans.PropertyChangeEvent))([PropertyChangeEvent](http://docs.google.com/java/beans/PropertyChangeEvent.html) e) |
| protected  void | [**updateViewport**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#updateViewport(java.beans.PropertyChangeEvent))([PropertyChangeEvent](http://docs.google.com/java/beans/PropertyChangeEvent.html) e) |

| **Methods inherited from class javax.swing.plaf.**[**ComponentUI**](http://docs.google.com/javax/swing/plaf/ComponentUI.html) |
| --- |
| [contains](http://docs.google.com/javax/swing/plaf/ComponentUI.html#contains(javax.swing.JComponent,%20int,%20int)), [getAccessibleChild](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getAccessibleChild(javax.swing.JComponent,%20int)), [getAccessibleChildrenCount](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getAccessibleChildrenCount(javax.swing.JComponent)), [getMinimumSize](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getMinimumSize(javax.swing.JComponent)), [getPreferredSize](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getPreferredSize(javax.swing.JComponent)), [update](http://docs.google.com/javax/swing/plaf/ComponentUI.html#update(java.awt.Graphics,%20javax.swing.JComponent)) |

| **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** |
| --- |

### scrollpane

protected [JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) **scrollpane**

### vsbChangeListener

protected [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) **vsbChangeListener**

### hsbChangeListener

protected [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) **hsbChangeListener**

### viewportChangeListener

protected [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) **viewportChangeListener**

### spPropertyChangeListener

protected [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) **spPropertyChangeListener**

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

### BasicScrollPaneUI

public **BasicScrollPaneUI**()

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

### createUI

public static [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **createUI**([JComponent](http://docs.google.com/javax/swing/JComponent.html) x)

### paint

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

**Description copied from class:** [**ComponentUI**](http://docs.google.com/javax/swing/plaf/ComponentUI.html#paint(java.awt.Graphics,%20javax.swing.JComponent)) Paints the specified component appropriate for the look and feel. This method is invoked from the ComponentUI.update method when the specified component is being painted. Subclasses should override this method and use the specified Graphics object to render the content of the component.

**Overrides:**[paint](http://docs.google.com/javax/swing/plaf/ComponentUI.html#paint(java.awt.Graphics,%20javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**g - the Graphics context in which to paintc - the component being painted; this argument is often ignored, but might be used if the UI object is stateless and shared by multiple components**See Also:**[ComponentUI.update(java.awt.Graphics, javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#update(java.awt.Graphics,%20javax.swing.JComponent))

### getMaximumSize

public [Dimension](http://docs.google.com/java/awt/Dimension.html) **getMaximumSize**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)

**Description copied from class:** [**ComponentUI**](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getMaximumSize(javax.swing.JComponent)) Returns the specified component's maximum size appropriate for the look and feel. If null is returned, the maximum size will be calculated by the component's layout manager instead (this is the preferred approach for any component with a specific layout manager installed). The default implementation of this method invokes getPreferredSize and returns that value.

**Overrides:**[getMaximumSize](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getMaximumSize(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**c - the component whose maximum size is being queried; this argument is often ignored, but might be used if the UI object is stateless and shared by multiple components **Returns:**new Dimension(Short.MAX\_VALUE, Short.MAX\_VALUE)**See Also:**[JComponent.getMaximumSize()](http://docs.google.com/javax/swing/JComponent.html#getMaximumSize()), [LayoutManager2.maximumLayoutSize(java.awt.Container)](http://docs.google.com/java/awt/LayoutManager2.html#maximumLayoutSize(java.awt.Container))

### installDefaults

protected void **installDefaults**([JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) scrollpane)

### installListeners

protected void **installListeners**([JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) c)

### installKeyboardActions

protected void **installKeyboardActions**([JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) c)

### installUI

public void **installUI**([JComponent](http://docs.google.com/javax/swing/JComponent.html) x)

**Description copied from class:** [**ComponentUI**](http://docs.google.com/javax/swing/plaf/ComponentUI.html#installUI(javax.swing.JComponent)) Configures the specified component appropriate for the look and feel. This method is invoked when the ComponentUI instance is being installed as the UI delegate on the specified component. This method should completely configure the component for the look and feel, including the following:

1. Install any default property values for color, fonts, borders, icons, opacity, etc. on the component. Whenever possible, property values initialized by the client program should *not* be overridden.
2. Install a LayoutManager on the component if necessary.
3. Create/add any required sub-components to the component.
4. Create/install event listeners on the component.
5. Create/install a PropertyChangeListener on the component in order to detect and respond to component property changes appropriately.
6. Install keyboard UI (mnemonics, traversal, etc.) on the component.
7. Initialize any appropriate instance data.

**Overrides:**[installUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html#installUI(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**x - the component where this UI delegate is being installed**See Also:**[ComponentUI.uninstallUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#uninstallUI(javax.swing.JComponent)), [JComponent.setUI(javax.swing.plaf.ComponentUI)](http://docs.google.com/javax/swing/JComponent.html#setUI(javax.swing.plaf.ComponentUI)), [JComponent.updateUI()](http://docs.google.com/javax/swing/JComponent.html#updateUI())

### uninstallDefaults

protected void **uninstallDefaults**([JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) c)

### uninstallListeners

protected void **uninstallListeners**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)

### uninstallKeyboardActions

protected void **uninstallKeyboardActions**([JScrollPane](http://docs.google.com/javax/swing/JScrollPane.html) c)

### uninstallUI

public void **uninstallUI**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)

**Description copied from class:** [**ComponentUI**](http://docs.google.com/javax/swing/plaf/ComponentUI.html#uninstallUI(javax.swing.JComponent)) Reverses configuration which was done on the specified component during installUI. This method is invoked when this UIComponent instance is being removed as the UI delegate for the specified component. This method should undo the configuration performed in installUI, being careful to leave the JComponent instance in a clean state (no extraneous listeners, look-and-feel-specific property objects, etc.). This should include the following:

1. Remove any UI-set borders from the component.
2. Remove any UI-set layout managers on the component.
3. Remove any UI-added sub-components from the component.
4. Remove any UI-added event/property listeners from the component.
5. Remove any UI-installed keyboard UI from the component.
6. Nullify any allocated instance data objects to allow for GC.

**Overrides:**[uninstallUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html#uninstallUI(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**c - the component from which this UI delegate is being removed; this argument is often ignored, but might be used if the UI object is stateless and shared by multiple components**See Also:**[ComponentUI.installUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#installUI(javax.swing.JComponent)), [JComponent.updateUI()](http://docs.google.com/javax/swing/JComponent.html#updateUI())

### syncScrollPaneWithViewport

protected void **syncScrollPaneWithViewport**()

### getBaseline

public int **getBaseline**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c,  
 int width,  
 int height)

Returns the baseline.

**Overrides:**[getBaseline](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getBaseline(javax.swing.JComponent,%20int,%20int)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**c - JComponent baseline is being requested forwidth - the width to get the baseline forheight - the height to get the baseline for **Returns:**baseline or a value < 0 indicating there is no reasonable baseline **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if c is null [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if width or height is < 0**Since:** 1.6 **See Also:**[JComponent.getBaseline(int, int)](http://docs.google.com/javax/swing/JComponent.html#getBaseline(int,%20int))

### getBaselineResizeBehavior

public [Component.BaselineResizeBehavior](http://docs.google.com/java/awt/Component.BaselineResizeBehavior.html) **getBaselineResizeBehavior**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)

Returns an enum indicating how the baseline of the component changes as the size changes.

**Overrides:**[getBaselineResizeBehavior](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getBaselineResizeBehavior(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**c - JComponent to return baseline resize behavior for **Returns:**an enum indicating how the baseline changes as the component size changes **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if c is null**Since:** 1.6 **See Also:**[JComponent.getBaseline(int, int)](http://docs.google.com/javax/swing/JComponent.html#getBaseline(int,%20int))

### createViewportChangeListener

protected [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) **createViewportChangeListener**()

### createHSBChangeListener

protected [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) **createHSBChangeListener**()

### createVSBChangeListener

protected [ChangeListener](http://docs.google.com/javax/swing/event/ChangeListener.html) **createVSBChangeListener**()

### createMouseWheelListener

protected [MouseWheelListener](http://docs.google.com/java/awt/event/MouseWheelListener.html) **createMouseWheelListener**()

Creates an instance of MouseWheelListener, which is added to the JScrollPane by installUI(). The returned MouseWheelListener is used to handle mouse wheel-driven scrolling.

**Returns:**MouseWheelListener which implements wheel-driven scrolling**Since:** 1.4 **See Also:**[installUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#installUI(javax.swing.JComponent)), [BasicScrollPaneUI.MouseWheelHandler](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.MouseWheelHandler.html)

### updateScrollBarDisplayPolicy

protected void **updateScrollBarDisplayPolicy**([PropertyChangeEvent](http://docs.google.com/java/beans/PropertyChangeEvent.html) e)

### updateViewport

protected void **updateViewport**([PropertyChangeEvent](http://docs.google.com/java/beans/PropertyChangeEvent.html) e)

### updateRowHeader

protected void **updateRowHeader**([PropertyChangeEvent](http://docs.google.com/java/beans/PropertyChangeEvent.html) e)

### updateColumnHeader

protected void **updateColumnHeader**([PropertyChangeEvent](http://docs.google.com/java/beans/PropertyChangeEvent.html) e)

### createPropertyChangeListener

protected [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) **createPropertyChangeListener**()

Creates an instance of PropertyChangeListener that's added to the JScrollPane by installUI(). Subclasses can override this method to return a custom PropertyChangeListener, e.g.

class MyScrollPaneUI extends BasicScrollPaneUI {  
 protected PropertyChangeListener **createPropertyChangeListener**() {  
 return new MyPropertyChangeListener();  
 }  
 public class MyPropertyChangeListener extends PropertyChangeListener {  
 public void propertyChange(PropertyChangeEvent e) {  
 if (e.getPropertyName().equals("viewport")) {  
 // do some extra work when the viewport changes  
 }  
 super.propertyChange(e);  
 }  
 }  
 }

**See Also:**[PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html), [installUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.html#installUI(javax.swing.JComponent))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BasicScrollPaneUI.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/plaf/basic/BasicScrollBarUI.TrackListener.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollPaneUI.HSBChangeListener.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/basic/BasicScrollPaneUI.html)    [**NO FRAMES**](http://docs.google.com/BasicScrollPaneUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2et92p0) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: [FIELD](#17dp8vu) | [CONSTR](#44sinio) | [METHOD](#z337ya) |

[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).