| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BasicSpinnerUI.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/BasicSliderUI.TrackListener.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/basic/BasicSpinnerUI.html)    [**NO FRAMES**](http://docs.google.com/BasicSpinnerUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#4d34og8) | [CONSTR](#17dp8vu) | [METHOD](#26in1rg) |

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

Class BasicSpinnerUI

[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.SpinnerUI](http://docs.google.com/javax/swing/plaf/SpinnerUI.html)  
 **javax.swing.plaf.basic.BasicSpinnerUI**

public class **BasicSpinnerUI**extends [SpinnerUI](http://docs.google.com/javax/swing/plaf/SpinnerUI.html)

The default Spinner UI delegate.

**Since:** 1.4

| **Field Summary** | |
| --- | --- |
| protected  [JSpinner](http://docs.google.com/javax/swing/JSpinner.html) | [**spinner**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#spinner)            The spinner that we're a UI delegate for. |

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

| **Method Summary** | |
| --- | --- |
| protected  [JComponent](http://docs.google.com/javax/swing/JComponent.html) | [**createEditor**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createEditor())()            This method is called by installUI to get the editor component of the JSpinner. |
| protected  [LayoutManager](http://docs.google.com/java/awt/LayoutManager.html) | [**createLayout**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createLayout())()            Create a LayoutManager that manages the editor, nextButton, and previousButton children of the JSpinner. |
| protected  [Component](http://docs.google.com/java/awt/Component.html) | [**createNextButton**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createNextButton())()            Create a component that will replace the spinner models value with the object returned by spinner.getNextValue. |
| protected  [Component](http://docs.google.com/java/awt/Component.html) | [**createPreviousButton**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createPreviousButton())()            Create a component that will replace the spinner models value with the object returned by spinner.getPreviousValue. |
| protected  [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) | [**createPropertyChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createPropertyChangeListener())()            Create a PropertyChangeListener that can be added to the JSpinner itself. |
| static [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) | [**createUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Returns a new instance of BasicSpinnerUI. |
| int | [**getBaseline**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.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/BasicSpinnerUI.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. |
| protected  void | [**installDefaults**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installDefaults())()            Initialize the JSpinner border, foreground, and background, properties based on the corresponding "Spinner.\*" properties from defaults table. |
| protected  void | [**installKeyboardActions**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installKeyboardActions())()            Installs the keyboard Actions onto the JSpinner. |
| protected  void | [**installListeners**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installListeners())()            Initializes PropertyChangeListener with a shared object that delegates interesting PropertyChangeEvents to protected methods. |
| protected  void | [**installNextButtonListeners**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installNextButtonListeners(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) c)            Installs the necessary listeners on the next button, c, to update the JSpinner in response to a user gesture. |
| protected  void | [**installPreviousButtonListeners**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installPreviousButtonListeners(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) c)            Installs the necessary listeners on the previous button, c, to update the JSpinner in response to a user gesture. |
| void | [**installUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Calls installDefaults, installListeners, and then adds the components returned by createNextButton, createPreviousButton, and createEditor. |
| protected  void | [**replaceEditor**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#replaceEditor(javax.swing.JComponent,%20javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) oldEditor, [JComponent](http://docs.google.com/javax/swing/JComponent.html) newEditor)            Called by the PropertyChangeListener when the JSpinner editor property changes. |
| protected  void | [**uninstallDefaults**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#uninstallDefaults())()            Sets the JSpinner's layout manager to null. |
| protected  void | [**uninstallListeners**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#uninstallListeners())()            Removes the PropertyChangeListener added by installListeners. |
| void | [**uninstallUI**](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#uninstallUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Calls uninstallDefaults, uninstallListeners, and then removes all of the spinners children. |

| **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)), [getMaximumSize](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getMaximumSize(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)), [paint](http://docs.google.com/javax/swing/plaf/ComponentUI.html#paint(java.awt.Graphics,%20javax.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** |
| --- |

### spinner

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

The spinner that we're a UI delegate for. Initialized by the installUI method, and reset to null by uninstallUI.

**See Also:**[installUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installUI(javax.swing.JComponent)), [uninstallUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#uninstallUI(javax.swing.JComponent))

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

### BasicSpinnerUI

public **BasicSpinnerUI**()

| **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) c)

Returns a new instance of BasicSpinnerUI. SpinnerListUI delegates are allocated one per JSpinner.

**Parameters:**c - the JSpinner (not used) **Returns:**a new BasicSpinnerUI object**See Also:**[ComponentUI.createUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#createUI(javax.swing.JComponent))

### installUI

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

Calls installDefaults, installListeners, and then adds the components returned by createNextButton, createPreviousButton, and createEditor.

**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:**c - the JSpinner**See Also:**[installDefaults()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installDefaults()), [installListeners()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installListeners()), [createNextButton()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createNextButton()), [createPreviousButton()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createPreviousButton()), [createEditor()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createEditor())

### uninstallUI

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

Calls uninstallDefaults, uninstallListeners, and then removes all of the spinners children.

**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 JSpinner (not used)**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())

### installListeners

protected void **installListeners**()

Initializes PropertyChangeListener with a shared object that delegates interesting PropertyChangeEvents to protected methods.

This method is called by installUI.

**See Also:**[replaceEditor(javax.swing.JComponent, javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#replaceEditor(javax.swing.JComponent,%20javax.swing.JComponent)), [uninstallListeners()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#uninstallListeners())

### uninstallListeners

protected void **uninstallListeners**()

Removes the PropertyChangeListener added by installListeners.

This method is called by uninstallUI.

**See Also:**[installListeners()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installListeners())

### installDefaults

protected void **installDefaults**()

Initialize the JSpinner border, foreground, and background, properties based on the corresponding "Spinner.\*" properties from defaults table. The JSpinners layout is set to the value returned by createLayout. This method is called by installUI.

**See Also:**[uninstallDefaults()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#uninstallDefaults()), [installUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installUI(javax.swing.JComponent)), [createLayout()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createLayout()), [LookAndFeel.installBorder(javax.swing.JComponent, java.lang.String)](http://docs.google.com/javax/swing/LookAndFeel.html#installBorder(javax.swing.JComponent,%20java.lang.String)), [LookAndFeel.installColors(javax.swing.JComponent, java.lang.String, java.lang.String)](http://docs.google.com/javax/swing/LookAndFeel.html#installColors(javax.swing.JComponent,%20java.lang.String,%20java.lang.String))

### uninstallDefaults

protected void **uninstallDefaults**()

Sets the JSpinner's layout manager to null. This method is called by uninstallUI.

**See Also:**[installDefaults()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installDefaults()), [uninstallUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#uninstallUI(javax.swing.JComponent))

### installNextButtonListeners

protected void **installNextButtonListeners**([Component](http://docs.google.com/java/awt/Component.html) c)

Installs the necessary listeners on the next button, c, to update the JSpinner in response to a user gesture.

**Parameters:**c - Component to install the listeners on **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if c is null.**Since:** 1.5 **See Also:**[createNextButton()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createNextButton())

### installPreviousButtonListeners

protected void **installPreviousButtonListeners**([Component](http://docs.google.com/java/awt/Component.html) c)

Installs the necessary listeners on the previous button, c, to update the JSpinner in response to a user gesture.

**Parameters:**c - Component to install the listeners on. **Throws:** [NullPointerException](http://docs.google.com/java/lang/NullPointerException.html) - if c is null.**Since:** 1.5 **See Also:**[createPreviousButton()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createPreviousButton())

### createLayout

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

Create a LayoutManager that manages the editor, nextButton, and previousButton children of the JSpinner. These three children must be added with a constraint that identifies their role: "Editor", "Next", and "Previous". The default layout manager can handle the absence of any of these children.

**Returns:**a LayoutManager for the editor, next button, and previous button.**See Also:**[createNextButton()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createNextButton()), [createPreviousButton()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createPreviousButton()), [createEditor()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createEditor())

### createPropertyChangeListener

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

Create a PropertyChangeListener that can be added to the JSpinner itself. Typically, this listener will call replaceEditor when the "editor" property changes, since it's the SpinnerUI's responsibility to add the editor to the JSpinner (and remove the old one). This method is called by installListeners.

**Returns:**A PropertyChangeListener for the JSpinner itself**See Also:**[installListeners()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installListeners())

### createPreviousButton

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

Create a component that will replace the spinner models value with the object returned by spinner.getPreviousValue. By default the previousButton is a JButton. This method invokes installPreviousButtonListeners to install the necessary listeners to update the JSpinner's model in response to a user gesture. If a previousButton isn't needed (in a subclass) then override this method to return null.

**Returns:**a component that will replace the spinners model with the next value in the sequence, or null**See Also:**[installUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installUI(javax.swing.JComponent)), [createNextButton()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createNextButton()), [installPreviousButtonListeners(java.awt.Component)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installPreviousButtonListeners(java.awt.Component))

### createNextButton

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

Create a component that will replace the spinner models value with the object returned by spinner.getNextValue. By default the nextButton is a JButton who's ActionListener updates it's JSpinner ancestors model. If a nextButton isn't needed (in a subclass) then override this method to return null.

**Returns:**a component that will replace the spinners model with the next value in the sequence, or null**See Also:**[installUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installUI(javax.swing.JComponent)), [createPreviousButton()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createPreviousButton()), [installNextButtonListeners(java.awt.Component)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installNextButtonListeners(java.awt.Component))

### createEditor

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

This method is called by installUI to get the editor component of the JSpinner. By default it just returns JSpinner.getEditor(). Subclasses can override createEditor to return a component that contains the spinner's editor or null, if they're going to handle adding the editor to the JSpinner in an installUI override.

Typically this method would be overridden to wrap the editor with a container with a custom border, since one can't assume that the editors border can be set directly.

The replaceEditor method is called when the spinners editor is changed with JSpinner.setEditor. If you've overriden this method, then you'll probably want to override replaceEditor as well.

**Returns:**the JSpinners editor JComponent, spinner.getEditor() by default**See Also:**[installUI(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#installUI(javax.swing.JComponent)), [replaceEditor(javax.swing.JComponent, javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#replaceEditor(javax.swing.JComponent,%20javax.swing.JComponent)), [JSpinner.getEditor()](http://docs.google.com/javax/swing/JSpinner.html#getEditor())

### replaceEditor

protected void **replaceEditor**([JComponent](http://docs.google.com/javax/swing/JComponent.html) oldEditor,  
 [JComponent](http://docs.google.com/javax/swing/JComponent.html) newEditor)

Called by the PropertyChangeListener when the JSpinner editor property changes. It's the responsibility of this method to remove the old editor and add the new one. By default this operation is just:

spinner.remove(oldEditor);  
 spinner.add(newEditor, "Editor");

The implementation of replaceEditor should be coordinated with the createEditor method.

**See Also:**[createEditor()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createEditor()), [createPropertyChangeListener()](http://docs.google.com/javax/swing/plaf/basic/BasicSpinnerUI.html#createPropertyChangeListener())

### installKeyboardActions

protected void **installKeyboardActions**()

Installs the keyboard Actions onto the JSpinner.

**Since:** 1.5

### 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))

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BasicSpinnerUI.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/BasicSliderUI.TrackListener.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/basic/BasicSpinnerUI.html)    [**NO FRAMES**](http://docs.google.com/BasicSpinnerUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#4d34og8) | [CONSTR](#17dp8vu) | [METHOD](#26in1rg) |

[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).