| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BasicDesktopPaneUI.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/BasicDesktopIconUI.MouseInputHandler.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.CloseAction.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/basic/BasicDesktopPaneUI.html)    [**NO FRAMES**](http://docs.google.com/BasicDesktopPaneUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: [FIELD](#2s8eyo1) | [CONSTR](#2jxsxqh) | [METHOD](#3j2qqm3) |

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

Class BasicDesktopPaneUI

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

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

Basic L&F for a desktop.

| **Nested Class Summary** | |
| --- | --- |
| protected  class | [**BasicDesktopPaneUI.CloseAction**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.CloseAction.html)            Handles closing an internal frame. |
| protected  class | [**BasicDesktopPaneUI.MaximizeAction**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.MaximizeAction.html)            Handles maximizing an internal frame. |
| protected  class | [**BasicDesktopPaneUI.MinimizeAction**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.MinimizeAction.html)            Handles minimizing an internal frame. |
| protected  class | [**BasicDesktopPaneUI.NavigateAction**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.NavigateAction.html)            Handles navigating to the next internal frame. |
| protected  class | [**BasicDesktopPaneUI.OpenAction**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.OpenAction.html)            Handles restoring a minimized or maximized internal frame. |

| **Field Summary** | |
| --- | --- |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**closeKey**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#closeKey)  **Deprecated.** *As of 1.3.* |
| protected  [JDesktopPane](http://docs.google.com/javax/swing/JDesktopPane.html) | [**desktop**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#desktop) |
| protected  [DesktopManager](http://docs.google.com/javax/swing/DesktopManager.html) | [**desktopManager**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#desktopManager) |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**maximizeKey**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#maximizeKey)  **Deprecated.** *As of 1.3.* |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**minimizeKey**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#minimizeKey)  **Deprecated.** *As of 1.3.* |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**navigateKey**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#navigateKey)  **Deprecated.** *As of 1.3.* |
| protected  [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) | [**navigateKey2**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#navigateKey2)  **Deprecated.** *As of 1.3.* |

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

| **Method Summary** | |
| --- | --- |
| protected  [PropertyChangeListener](http://docs.google.com/java/beans/PropertyChangeListener.html) | [**createPropertyChangeListener**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#createPropertyChangeListener())()            Returns the PropertyChangeListener to install on the JDesktopPane. |
| static [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) | [**createUI**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#createUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c) |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getMaximumSize**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.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. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getMinimumSize**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#getMinimumSize(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Returns the specified component's minimum size appropriate for the look and feel. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getPreferredSize**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#getPreferredSize(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Returns the specified component's preferred size appropriate for the look and feel. |
| protected  void | [**installDefaults**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#installDefaults())() |
| protected  void | [**installDesktopManager**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#installDesktopManager())() |
| protected  void | [**installKeyboardActions**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#installKeyboardActions())() |
| protected  void | [**installListeners**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#installListeners())()            Installs the PropertyChangeListener returned from createPropertyChangeListener on the JDesktopPane. |
| void | [**installUI**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#installUI(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Configures the specified component appropriate for the look and feel. |
| void | [**paint**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.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 | [**registerKeyboardActions**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#registerKeyboardActions())() |
| protected  void | [**uninstallDefaults**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#uninstallDefaults())() |
| protected  void | [**uninstallDesktopManager**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#uninstallDesktopManager())() |
| protected  void | [**uninstallKeyboardActions**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#uninstallKeyboardActions())() |
| protected  void | [**uninstallListeners**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#uninstallListeners())()            Uninstalls the PropertyChangeListener returned from createPropertyChangeListener from the JDesktopPane. |
| void | [**uninstallUI**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.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 | [**unregisterKeyboardActions**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#unregisterKeyboardActions())() |

| **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)), [getBaseline](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getBaseline(javax.swing.JComponent,%20int,%20int)), [getBaselineResizeBehavior](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getBaselineResizeBehavior(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** |
| --- |

### desktop

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

### desktopManager

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

### minimizeKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **minimizeKey**

**Deprecated.** *As of 1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### maximizeKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **maximizeKey**

**Deprecated.** *As of 1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### closeKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **closeKey**

**Deprecated.** *As of 1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### navigateKey

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **navigateKey**

**Deprecated.** *As of 1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

### navigateKey2

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
protected [KeyStroke](http://docs.google.com/javax/swing/KeyStroke.html) **navigateKey2**

**Deprecated.** *As of 1.3.*As of Java 2 platform v1.3 this previously undocumented field is no longer used. Key bindings are now defined by the LookAndFeel, please refer to the key bindings specification for further details.

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

### BasicDesktopPaneUI

public **BasicDesktopPaneUI**()

| **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)

### installUI

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

**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:**c - 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())

### 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())

### installDefaults

protected void **installDefaults**()

### uninstallDefaults

protected void **uninstallDefaults**()

### installListeners

protected void **installListeners**()

Installs the PropertyChangeListener returned from createPropertyChangeListener on the JDesktopPane.

**Since:** 1.5 **See Also:**[createPropertyChangeListener()](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#createPropertyChangeListener())

### uninstallListeners

protected void **uninstallListeners**()

Uninstalls the PropertyChangeListener returned from createPropertyChangeListener from the JDesktopPane.

**Since:** 1.5 **See Also:**[createPropertyChangeListener()](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.html#createPropertyChangeListener())

### installDesktopManager

protected void **installDesktopManager**()

### uninstallDesktopManager

protected void **uninstallDesktopManager**()

### installKeyboardActions

protected void **installKeyboardActions**()

### registerKeyboardActions

protected void **registerKeyboardActions**()

### unregisterKeyboardActions

protected void **unregisterKeyboardActions**()

### uninstallKeyboardActions

protected void **uninstallKeyboardActions**()

### 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))

### getPreferredSize

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

**Description copied from class:** [**ComponentUI**](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getPreferredSize(javax.swing.JComponent)) Returns the specified component's preferred size appropriate for the look and feel. If null is returned, the preferred 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 returns null.

**Overrides:**[getPreferredSize](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getPreferredSize(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**c - the component whose preferred size is being queried; this argument is often ignored, but might be used if the UI object is stateless and shared by multiple components**See Also:**[JComponent.getPreferredSize()](http://docs.google.com/javax/swing/JComponent.html#getPreferredSize()), [LayoutManager.preferredLayoutSize(java.awt.Container)](http://docs.google.com/java/awt/LayoutManager.html#preferredLayoutSize(java.awt.Container))

### getMinimumSize

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

**Description copied from class:** [**ComponentUI**](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getMinimumSize(javax.swing.JComponent)) Returns the specified component's minimum size appropriate for the look and feel. If null is returned, the minimum 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:**[getMinimumSize](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getMinimumSize(javax.swing.JComponent)) in class [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html) **Parameters:**c - the component whose minimum 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:**a Dimension object or null**See Also:**[JComponent.getMinimumSize()](http://docs.google.com/javax/swing/JComponent.html#getMinimumSize()), [LayoutManager.minimumLayoutSize(java.awt.Container)](http://docs.google.com/java/awt/LayoutManager.html#minimumLayoutSize(java.awt.Container)), [ComponentUI.getPreferredSize(javax.swing.JComponent)](http://docs.google.com/javax/swing/plaf/ComponentUI.html#getPreferredSize(javax.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:**a Dimension object or null**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))

### createPropertyChangeListener

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

Returns the PropertyChangeListener to install on the JDesktopPane.

**Returns:**The PropertyChangeListener that will be added to track changes in the desktop pane.**Since:** 1.5

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/BasicDesktopPaneUI.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/BasicDesktopIconUI.MouseInputHandler.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopPaneUI.CloseAction.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/plaf/basic/BasicDesktopPaneUI.html)    [**NO FRAMES**](http://docs.google.com/BasicDesktopPaneUI.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2et92p0) | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: [FIELD](#2s8eyo1) | [CONSTR](#2jxsxqh) | [METHOD](#3j2qqm3) |

[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).