| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/JLayeredPane.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/JLabel.AccessibleJLabel.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/JLayeredPane.AccessibleJLayeredPane.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/JLayeredPane.html)    [**NO FRAMES**](http://docs.google.com/JLayeredPane.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#1t3h5sf) | [CONSTR](#3rdcrjn) | [METHOD](#26in1rg) | DETAIL: [FIELD](#2jxsxqh) | [CONSTR](#2bn6wsx) | [METHOD](#3as4poj) |

## **javax.swing**

Class JLayeredPane

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.awt.Component](http://docs.google.com/java/awt/Component.html)  
 [java.awt.Container](http://docs.google.com/java/awt/Container.html)  
 [javax.swing.JComponent](http://docs.google.com/javax/swing/JComponent.html)  
 **javax.swing.JLayeredPane**

**All Implemented Interfaces:** [ImageObserver](http://docs.google.com/java/awt/image/ImageObserver.html), [MenuContainer](http://docs.google.com/java/awt/MenuContainer.html), [Serializable](http://docs.google.com/java/io/Serializable.html), [Accessible](http://docs.google.com/javax/accessibility/Accessible.html) **Direct Known Subclasses:** [JDesktopPane](http://docs.google.com/javax/swing/JDesktopPane.html)

public class **JLayeredPane**extends [JComponent](http://docs.google.com/javax/swing/JComponent.html)implements [Accessible](http://docs.google.com/javax/accessibility/Accessible.html)

JLayeredPane adds depth to a JFC/Swing container, allowing components to overlap each other when needed. An Integer object specifies each component's depth in the container, where higher-numbered components sit "on top" of other components. For task-oriented documentation and examples of using layered panes see [How to Use a Layered Pane](http://java.sun.com/docs/books/tutorial/uiswing/components/layeredpane.html), a section in *The Java Tutorial*.

|  |
| --- |

For convenience, JLayeredPane divides the depth-range into several different layers. Putting a component into one of those layers makes it easy to ensure that components overlap properly, without having to worry about specifying numbers for specific depths:

DEFAULT\_LAYER The standard layer, where most components go. This the bottommost layer. PALETTE\_LAYER The palette layer sits over the default layer. Useful for floating toolbars and palettes, so they can be positioned above other components. MODAL\_LAYER The layer used for modal dialogs. They will appear on top of any toolbars, palettes, or standard components in the container. POPUP\_LAYER The popup layer displays above dialogs. That way, the popup windows associated with combo boxes, tooltips, and other help text will appear above the component, palette, or dialog that generated them. DRAG\_LAYER When dragging a component, reassigning it to the drag layer ensures that it is positioned over every other component in the container. When finished dragging, it can be reassigned to its normal layer. The JLayeredPane methods moveToFront(Component), moveToBack(Component) and setPosition can be used to reposition a component within its layer. The setLayer method can also be used to change the component's current layer.

## Details

JLayeredPane manages its list of children like Container, but allows for the definition of a several layers within itself. Children in the same layer are managed exactly like the normal Container object, with the added feature that when children components overlap, children in higher layers display above the children in lower layers.

Each layer is a distinct integer number. The layer attribute can be set on a Component by passing an Integer object during the add call.

For example:

layeredPane.add(child, JLayeredPane.DEFAULT\_LAYER);  
 or  
 layeredPane.add(child, new Integer(10));

The layer attribute can also be set on a Component by calling

layeredPaneParent.setLayer(child, 10)

on the JLayeredPane that is the parent of component. The layer should be set *before* adding the child to the parent.

Higher number layers display above lower number layers. So, using numbers for the layers and letters for individual components, a representative list order would look like this:

5a, 5b, 5c, 2a, 2b, 2c, 1a

where the leftmost components are closest to the top of the display.

A component can be moved to the top or bottom position within its layer by calling moveToFront or moveToBack.

The position of a component within a layer can also be specified directly. Valid positions range from 0 up to one less than the number of components in that layer. A value of -1 indicates the bottommost position. A value of 0 indicates the topmost position. Unlike layer numbers, higher position values are *lower* in the display.

**Note:** This sequence (defined by java.awt.Container) is the reverse of the layer numbering sequence. Usually though, you will use moveToFront, moveToBack, and setLayer.Here are some examples using the method add(Component, layer, position): Calling add(5x, 5, -1) results in:

5a, 5b, 5c, 5x, 2a, 2b, 2c, 1a

Calling add(5z, 5, 2) results in:

5a, 5b, 5z, 5c, 5x, 2a, 2b, 2c, 1a

Calling add(3a, 3, 7) results in:

5a, 5b, 5z, 5c, 5x, 3a, 2a, 2b, 2c, 1a

Using normal paint/event mechanics results in 1a appearing at the bottom and 5a being above all other components.

**Note:** that these layers are simply a logical construct and LayoutManagers will affect all child components of this container without regard for layer settings.

**Warning:** Swing is not thread safe. For more information see [Swing's Threading Policy](http://docs.google.com/package-summary.html#threading).

**Warning:** Serialized objects of this class will not be compatible with future Swing releases. The current serialization support is appropriate for short term storage or RMI between applications running the same version of Swing. As of 1.4, support for long term storage of all JavaBeansTM has been added to the java.beans package. Please see [XMLEncoder](http://docs.google.com/java/beans/XMLEncoder.html).

| **Nested Class Summary** | |
| --- | --- |
| protected  class | [**JLayeredPane.AccessibleJLayeredPane**](http://docs.google.com/javax/swing/JLayeredPane.AccessibleJLayeredPane.html)            This class implements accessibility support for the JLayeredPane class. |

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

| **Nested classes/interfaces inherited from class java.awt.**[**Container**](http://docs.google.com/java/awt/Container.html) |
| --- |
| [Container.AccessibleAWTContainer](http://docs.google.com/java/awt/Container.AccessibleAWTContainer.html) |

| **Nested classes/interfaces inherited from class java.awt.**[**Component**](http://docs.google.com/java/awt/Component.html) |
| --- |
| [Component.AccessibleAWTComponent](http://docs.google.com/java/awt/Component.AccessibleAWTComponent.html), [Component.BaselineResizeBehavior](http://docs.google.com/java/awt/Component.BaselineResizeBehavior.html), [Component.BltBufferStrategy](http://docs.google.com/java/awt/Component.BltBufferStrategy.html), [Component.FlipBufferStrategy](http://docs.google.com/java/awt/Component.FlipBufferStrategy.html) |

| **Field Summary** | |
| --- | --- |
| static [Integer](http://docs.google.com/java/lang/Integer.html) | [**DEFAULT\_LAYER**](http://docs.google.com/javax/swing/JLayeredPane.html#DEFAULT_LAYER)            Convenience object defining the Default layer. |
| static [Integer](http://docs.google.com/java/lang/Integer.html) | [**DRAG\_LAYER**](http://docs.google.com/javax/swing/JLayeredPane.html#DRAG_LAYER)            Convenience object defining the Drag layer. |
| static [Integer](http://docs.google.com/java/lang/Integer.html) | [**FRAME\_CONTENT\_LAYER**](http://docs.google.com/javax/swing/JLayeredPane.html#FRAME_CONTENT_LAYER)            Convenience object defining the Frame Content layer. |
| static [String](http://docs.google.com/java/lang/String.html) | [**LAYER\_PROPERTY**](http://docs.google.com/javax/swing/JLayeredPane.html#LAYER_PROPERTY)            Bound property |
| static [Integer](http://docs.google.com/java/lang/Integer.html) | [**MODAL\_LAYER**](http://docs.google.com/javax/swing/JLayeredPane.html#MODAL_LAYER)            Convenience object defining the Modal layer. |
| static [Integer](http://docs.google.com/java/lang/Integer.html) | [**PALETTE\_LAYER**](http://docs.google.com/javax/swing/JLayeredPane.html#PALETTE_LAYER)            Convenience object defining the Palette layer. |
| static [Integer](http://docs.google.com/java/lang/Integer.html) | [**POPUP\_LAYER**](http://docs.google.com/javax/swing/JLayeredPane.html#POPUP_LAYER)            Convenience object defining the Popup layer. |

| **Fields inherited from class javax.swing.**[**JComponent**](http://docs.google.com/javax/swing/JComponent.html) |
| --- |
| [accessibleContext](http://docs.google.com/javax/swing/JComponent.html#accessibleContext), [listenerList](http://docs.google.com/javax/swing/JComponent.html#listenerList), [TOOL\_TIP\_TEXT\_KEY](http://docs.google.com/javax/swing/JComponent.html#TOOL_TIP_TEXT_KEY), [ui](http://docs.google.com/javax/swing/JComponent.html#ui), [UNDEFINED\_CONDITION](http://docs.google.com/javax/swing/JComponent.html#UNDEFINED_CONDITION), [WHEN\_ANCESTOR\_OF\_FOCUSED\_COMPONENT](http://docs.google.com/javax/swing/JComponent.html#WHEN_ANCESTOR_OF_FOCUSED_COMPONENT), [WHEN\_FOCUSED](http://docs.google.com/javax/swing/JComponent.html#WHEN_FOCUSED), [WHEN\_IN\_FOCUSED\_WINDOW](http://docs.google.com/javax/swing/JComponent.html#WHEN_IN_FOCUSED_WINDOW) |

| **Fields inherited from class java.awt.**[**Component**](http://docs.google.com/java/awt/Component.html) |
| --- |
| [BOTTOM\_ALIGNMENT](http://docs.google.com/java/awt/Component.html#BOTTOM_ALIGNMENT), [CENTER\_ALIGNMENT](http://docs.google.com/java/awt/Component.html#CENTER_ALIGNMENT), [LEFT\_ALIGNMENT](http://docs.google.com/java/awt/Component.html#LEFT_ALIGNMENT), [RIGHT\_ALIGNMENT](http://docs.google.com/java/awt/Component.html#RIGHT_ALIGNMENT), [TOP\_ALIGNMENT](http://docs.google.com/java/awt/Component.html#TOP_ALIGNMENT) |

| **Fields inherited from interface java.awt.image.**[**ImageObserver**](http://docs.google.com/java/awt/image/ImageObserver.html) |
| --- |
| [ABORT](http://docs.google.com/java/awt/image/ImageObserver.html#ABORT), [ALLBITS](http://docs.google.com/java/awt/image/ImageObserver.html#ALLBITS), [ERROR](http://docs.google.com/java/awt/image/ImageObserver.html#ERROR), [FRAMEBITS](http://docs.google.com/java/awt/image/ImageObserver.html#FRAMEBITS), [HEIGHT](http://docs.google.com/java/awt/image/ImageObserver.html#HEIGHT), [PROPERTIES](http://docs.google.com/java/awt/image/ImageObserver.html#PROPERTIES), [SOMEBITS](http://docs.google.com/java/awt/image/ImageObserver.html#SOMEBITS), [WIDTH](http://docs.google.com/java/awt/image/ImageObserver.html#WIDTH) |

| **Constructor Summary** | |
| --- | --- |
| [**JLayeredPane**](http://docs.google.com/javax/swing/JLayeredPane.html#JLayeredPane())()            Create a new JLayeredPane |

| **Method Summary** | |
| --- | --- |
| protected  void | [**addImpl**](http://docs.google.com/javax/swing/JLayeredPane.html#addImpl(java.awt.Component,%20java.lang.Object,%20int))([Component](http://docs.google.com/java/awt/Component.html) comp, [Object](http://docs.google.com/java/lang/Object.html) constraints, int index)            Adds the specified component to this container at the specified index. |
| [AccessibleContext](http://docs.google.com/javax/accessibility/AccessibleContext.html) | [**getAccessibleContext**](http://docs.google.com/javax/swing/JLayeredPane.html#getAccessibleContext())()            Gets the AccessibleContext associated with this JLayeredPane. |
| int | [**getComponentCountInLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#getComponentCountInLayer(int))(int layer)            Returns the number of children currently in the specified layer. |
| [Component](http://docs.google.com/java/awt/Component.html)[] | [**getComponentsInLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#getComponentsInLayer(int))(int layer)            Returns an array of the components in the specified layer. |
| protected  [Hashtable](http://docs.google.com/java/util/Hashtable.html)<[Component](http://docs.google.com/java/awt/Component.html),[Integer](http://docs.google.com/java/lang/Integer.html)> | [**getComponentToLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#getComponentToLayer())()            Returns the hashtable that maps components to layers. |
| int | [**getIndexOf**](http://docs.google.com/javax/swing/JLayeredPane.html#getIndexOf(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) c)            Returns the index of the specified Component. |
| int | [**getLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#getLayer(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) c)            Returns the layer attribute for the specified Component. |
| static int | [**getLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#getLayer(javax.swing.JComponent))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)            Gets the layer property for a JComponent, it does not cause any side effects like setLayer(). |
| static [JLayeredPane](http://docs.google.com/javax/swing/JLayeredPane.html) | [**getLayeredPaneAbove**](http://docs.google.com/javax/swing/JLayeredPane.html#getLayeredPaneAbove(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) c)            Convenience method that returns the first JLayeredPane which contains the specified component. |
| protected  [Integer](http://docs.google.com/java/lang/Integer.html) | [**getObjectForLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#getObjectForLayer(int))(int layer)            Returns the Integer object associated with a specified layer. |
| int | [**getPosition**](http://docs.google.com/javax/swing/JLayeredPane.html#getPosition(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) c)            Get the relative position of the component within its layer. |
| int | [**highestLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#highestLayer())()            Returns the highest layer value from all current children. |
| protected  int | [**insertIndexForLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#insertIndexForLayer(int,%20int))(int layer, int position)            Primitive method that determines the proper location to insert a new child based on layer and position requests. |
| boolean | [**isOptimizedDrawingEnabled**](http://docs.google.com/javax/swing/JLayeredPane.html#isOptimizedDrawingEnabled())()            Returns false if components in the pane can overlap, which makes optimized drawing impossible. |
| int | [**lowestLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#lowestLayer())()            Returns the lowest layer value from all current children. |
| void | [**moveToBack**](http://docs.google.com/javax/swing/JLayeredPane.html#moveToBack(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) c)            Moves the component to the bottom of the components in its current layer (position -1). |
| void | [**moveToFront**](http://docs.google.com/javax/swing/JLayeredPane.html#moveToFront(java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) c)            Moves the component to the top of the components in its current layer (position 0). |
| void | [**paint**](http://docs.google.com/javax/swing/JLayeredPane.html#paint(java.awt.Graphics))([Graphics](http://docs.google.com/java/awt/Graphics.html) g)            Paints this JLayeredPane within the specified graphics context. |
| protected  [String](http://docs.google.com/java/lang/String.html) | [**paramString**](http://docs.google.com/javax/swing/JLayeredPane.html#paramString())()            Returns a string representation of this JLayeredPane. |
| static void | [**putLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#putLayer(javax.swing.JComponent,%20int))([JComponent](http://docs.google.com/javax/swing/JComponent.html) c, int layer)            Sets the layer property on a JComponent. |
| void | [**remove**](http://docs.google.com/javax/swing/JLayeredPane.html#remove(int))(int index)            Remove the indexed component from this pane. |
| void | [**removeAll**](http://docs.google.com/javax/swing/JLayeredPane.html#removeAll())()            Removes all the components from this container. |
| void | [**setLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#setLayer(java.awt.Component,%20int))([Component](http://docs.google.com/java/awt/Component.html) c, int layer)            Sets the layer attribute on the specified component, making it the bottommost component in that layer. |
| void | [**setLayer**](http://docs.google.com/javax/swing/JLayeredPane.html#setLayer(java.awt.Component,%20int,%20int))([Component](http://docs.google.com/java/awt/Component.html) c, int layer, int position)            Sets the layer attribute for the specified component and also sets its position within that layer. |
| void | [**setPosition**](http://docs.google.com/javax/swing/JLayeredPane.html#setPosition(java.awt.Component,%20int))([Component](http://docs.google.com/java/awt/Component.html) c, int position)            Moves the component to position within its current layer, where 0 is the topmost position within the layer and -1 is the bottommost position. |

| **Methods inherited from class javax.swing.**[**JComponent**](http://docs.google.com/javax/swing/JComponent.html) |
| --- |
| [addAncestorListener](http://docs.google.com/javax/swing/JComponent.html#addAncestorListener(javax.swing.event.AncestorListener)), [addNotify](http://docs.google.com/javax/swing/JComponent.html#addNotify()), [addVetoableChangeListener](http://docs.google.com/javax/swing/JComponent.html#addVetoableChangeListener(java.beans.VetoableChangeListener)), [computeVisibleRect](http://docs.google.com/javax/swing/JComponent.html#computeVisibleRect(java.awt.Rectangle)), [contains](http://docs.google.com/javax/swing/JComponent.html#contains(int,%20int)), [createToolTip](http://docs.google.com/javax/swing/JComponent.html#createToolTip()), [disable](http://docs.google.com/javax/swing/JComponent.html#disable()), [enable](http://docs.google.com/javax/swing/JComponent.html#enable()), [firePropertyChange](http://docs.google.com/javax/swing/JComponent.html#firePropertyChange(java.lang.String,%20boolean,%20boolean)), [firePropertyChange](http://docs.google.com/javax/swing/JComponent.html#firePropertyChange(java.lang.String,%20char,%20char)), [firePropertyChange](http://docs.google.com/javax/swing/JComponent.html#firePropertyChange(java.lang.String,%20int,%20int)), [fireVetoableChange](http://docs.google.com/javax/swing/JComponent.html#fireVetoableChange(java.lang.String,%20java.lang.Object,%20java.lang.Object)), [getActionForKeyStroke](http://docs.google.com/javax/swing/JComponent.html#getActionForKeyStroke(javax.swing.KeyStroke)), [getActionMap](http://docs.google.com/javax/swing/JComponent.html#getActionMap()), [getAlignmentX](http://docs.google.com/javax/swing/JComponent.html#getAlignmentX()), [getAlignmentY](http://docs.google.com/javax/swing/JComponent.html#getAlignmentY()), [getAncestorListeners](http://docs.google.com/javax/swing/JComponent.html#getAncestorListeners()), [getAutoscrolls](http://docs.google.com/javax/swing/JComponent.html#getAutoscrolls()), [getBaseline](http://docs.google.com/javax/swing/JComponent.html#getBaseline(int,%20int)), [getBaselineResizeBehavior](http://docs.google.com/javax/swing/JComponent.html#getBaselineResizeBehavior()), [getBorder](http://docs.google.com/javax/swing/JComponent.html#getBorder()), [getBounds](http://docs.google.com/javax/swing/JComponent.html#getBounds(java.awt.Rectangle)), [getClientProperty](http://docs.google.com/javax/swing/JComponent.html#getClientProperty(java.lang.Object)), [getComponentGraphics](http://docs.google.com/javax/swing/JComponent.html#getComponentGraphics(java.awt.Graphics)), [getComponentPopupMenu](http://docs.google.com/javax/swing/JComponent.html#getComponentPopupMenu()), [getConditionForKeyStroke](http://docs.google.com/javax/swing/JComponent.html#getConditionForKeyStroke(javax.swing.KeyStroke)), [getDebugGraphicsOptions](http://docs.google.com/javax/swing/JComponent.html#getDebugGraphicsOptions()), [getDefaultLocale](http://docs.google.com/javax/swing/JComponent.html#getDefaultLocale()), [getFontMetrics](http://docs.google.com/javax/swing/JComponent.html#getFontMetrics(java.awt.Font)), [getGraphics](http://docs.google.com/javax/swing/JComponent.html#getGraphics()), [getHeight](http://docs.google.com/javax/swing/JComponent.html#getHeight()), [getInheritsPopupMenu](http://docs.google.com/javax/swing/JComponent.html#getInheritsPopupMenu()), [getInputMap](http://docs.google.com/javax/swing/JComponent.html#getInputMap()), [getInputMap](http://docs.google.com/javax/swing/JComponent.html#getInputMap(int)), [getInputVerifier](http://docs.google.com/javax/swing/JComponent.html#getInputVerifier()), [getInsets](http://docs.google.com/javax/swing/JComponent.html#getInsets()), [getInsets](http://docs.google.com/javax/swing/JComponent.html#getInsets(java.awt.Insets)), [getListeners](http://docs.google.com/javax/swing/JComponent.html#getListeners(java.lang.Class)), [getLocation](http://docs.google.com/javax/swing/JComponent.html#getLocation(java.awt.Point)), [getMaximumSize](http://docs.google.com/javax/swing/JComponent.html#getMaximumSize()), [getMinimumSize](http://docs.google.com/javax/swing/JComponent.html#getMinimumSize()), [getNextFocusableComponent](http://docs.google.com/javax/swing/JComponent.html#getNextFocusableComponent()), [getPopupLocation](http://docs.google.com/javax/swing/JComponent.html#getPopupLocation(java.awt.event.MouseEvent)), [getPreferredSize](http://docs.google.com/javax/swing/JComponent.html#getPreferredSize()), [getRegisteredKeyStrokes](http://docs.google.com/javax/swing/JComponent.html#getRegisteredKeyStrokes()), [getRootPane](http://docs.google.com/javax/swing/JComponent.html#getRootPane()), [getSize](http://docs.google.com/javax/swing/JComponent.html#getSize(java.awt.Dimension)), [getToolTipLocation](http://docs.google.com/javax/swing/JComponent.html#getToolTipLocation(java.awt.event.MouseEvent)), [getToolTipText](http://docs.google.com/javax/swing/JComponent.html#getToolTipText()), [getToolTipText](http://docs.google.com/javax/swing/JComponent.html#getToolTipText(java.awt.event.MouseEvent)), [getTopLevelAncestor](http://docs.google.com/javax/swing/JComponent.html#getTopLevelAncestor()), [getTransferHandler](http://docs.google.com/javax/swing/JComponent.html#getTransferHandler()), [getUIClassID](http://docs.google.com/javax/swing/JComponent.html#getUIClassID()), [getVerifyInputWhenFocusTarget](http://docs.google.com/javax/swing/JComponent.html#getVerifyInputWhenFocusTarget()), [getVetoableChangeListeners](http://docs.google.com/javax/swing/JComponent.html#getVetoableChangeListeners()), [getVisibleRect](http://docs.google.com/javax/swing/JComponent.html#getVisibleRect()), [getWidth](http://docs.google.com/javax/swing/JComponent.html#getWidth()), [getX](http://docs.google.com/javax/swing/JComponent.html#getX()), [getY](http://docs.google.com/javax/swing/JComponent.html#getY()), [grabFocus](http://docs.google.com/javax/swing/JComponent.html#grabFocus()), [isDoubleBuffered](http://docs.google.com/javax/swing/JComponent.html#isDoubleBuffered()), [isLightweightComponent](http://docs.google.com/javax/swing/JComponent.html#isLightweightComponent(java.awt.Component)), [isManagingFocus](http://docs.google.com/javax/swing/JComponent.html#isManagingFocus()), [isOpaque](http://docs.google.com/javax/swing/JComponent.html#isOpaque()), [isPaintingForPrint](http://docs.google.com/javax/swing/JComponent.html#isPaintingForPrint()), [isPaintingTile](http://docs.google.com/javax/swing/JComponent.html#isPaintingTile()), [isRequestFocusEnabled](http://docs.google.com/javax/swing/JComponent.html#isRequestFocusEnabled()), [isValidateRoot](http://docs.google.com/javax/swing/JComponent.html#isValidateRoot()), [paintBorder](http://docs.google.com/javax/swing/JComponent.html#paintBorder(java.awt.Graphics)), [paintChildren](http://docs.google.com/javax/swing/JComponent.html#paintChildren(java.awt.Graphics)), [paintComponent](http://docs.google.com/javax/swing/JComponent.html#paintComponent(java.awt.Graphics)), [paintImmediately](http://docs.google.com/javax/swing/JComponent.html#paintImmediately(int,%20int,%20int,%20int)), [paintImmediately](http://docs.google.com/javax/swing/JComponent.html#paintImmediately(java.awt.Rectangle)), [print](http://docs.google.com/javax/swing/JComponent.html#print(java.awt.Graphics)), [printAll](http://docs.google.com/javax/swing/JComponent.html#printAll(java.awt.Graphics)), [printBorder](http://docs.google.com/javax/swing/JComponent.html#printBorder(java.awt.Graphics)), [printChildren](http://docs.google.com/javax/swing/JComponent.html#printChildren(java.awt.Graphics)), [printComponent](http://docs.google.com/javax/swing/JComponent.html#printComponent(java.awt.Graphics)), [processComponentKeyEvent](http://docs.google.com/javax/swing/JComponent.html#processComponentKeyEvent(java.awt.event.KeyEvent)), [processKeyBinding](http://docs.google.com/javax/swing/JComponent.html#processKeyBinding(javax.swing.KeyStroke,%20java.awt.event.KeyEvent,%20int,%20boolean)), [processKeyEvent](http://docs.google.com/javax/swing/JComponent.html#processKeyEvent(java.awt.event.KeyEvent)), [processMouseEvent](http://docs.google.com/javax/swing/JComponent.html#processMouseEvent(java.awt.event.MouseEvent)), [processMouseMotionEvent](http://docs.google.com/javax/swing/JComponent.html#processMouseMotionEvent(java.awt.event.MouseEvent)), [putClientProperty](http://docs.google.com/javax/swing/JComponent.html#putClientProperty(java.lang.Object,%20java.lang.Object)), [registerKeyboardAction](http://docs.google.com/javax/swing/JComponent.html#registerKeyboardAction(java.awt.event.ActionListener,%20javax.swing.KeyStroke,%20int)), [registerKeyboardAction](http://docs.google.com/javax/swing/JComponent.html#registerKeyboardAction(java.awt.event.ActionListener,%20java.lang.String,%20javax.swing.KeyStroke,%20int)), [removeAncestorListener](http://docs.google.com/javax/swing/JComponent.html#removeAncestorListener(javax.swing.event.AncestorListener)), [removeNotify](http://docs.google.com/javax/swing/JComponent.html#removeNotify()), [removeVetoableChangeListener](http://docs.google.com/javax/swing/JComponent.html#removeVetoableChangeListener(java.beans.VetoableChangeListener)), [repaint](http://docs.google.com/javax/swing/JComponent.html#repaint(long,%20int,%20int,%20int,%20int)), [repaint](http://docs.google.com/javax/swing/JComponent.html#repaint(java.awt.Rectangle)), [requestDefaultFocus](http://docs.google.com/javax/swing/JComponent.html#requestDefaultFocus()), [requestFocus](http://docs.google.com/javax/swing/JComponent.html#requestFocus()), [requestFocus](http://docs.google.com/javax/swing/JComponent.html#requestFocus(boolean)), [requestFocusInWindow](http://docs.google.com/javax/swing/JComponent.html#requestFocusInWindow()), [requestFocusInWindow](http://docs.google.com/javax/swing/JComponent.html#requestFocusInWindow(boolean)), [resetKeyboardActions](http://docs.google.com/javax/swing/JComponent.html#resetKeyboardActions()), [reshape](http://docs.google.com/javax/swing/JComponent.html#reshape(int,%20int,%20int,%20int)), [revalidate](http://docs.google.com/javax/swing/JComponent.html#revalidate()), [scrollRectToVisible](http://docs.google.com/javax/swing/JComponent.html#scrollRectToVisible(java.awt.Rectangle)), [setActionMap](http://docs.google.com/javax/swing/JComponent.html#setActionMap(javax.swing.ActionMap)), [setAlignmentX](http://docs.google.com/javax/swing/JComponent.html#setAlignmentX(float)), [setAlignmentY](http://docs.google.com/javax/swing/JComponent.html#setAlignmentY(float)), [setAutoscrolls](http://docs.google.com/javax/swing/JComponent.html#setAutoscrolls(boolean)), [setBackground](http://docs.google.com/javax/swing/JComponent.html#setBackground(java.awt.Color)), [setBorder](http://docs.google.com/javax/swing/JComponent.html#setBorder(javax.swing.border.Border)), [setComponentPopupMenu](http://docs.google.com/javax/swing/JComponent.html#setComponentPopupMenu(javax.swing.JPopupMenu)), [setDebugGraphicsOptions](http://docs.google.com/javax/swing/JComponent.html#setDebugGraphicsOptions(int)), [setDefaultLocale](http://docs.google.com/javax/swing/JComponent.html#setDefaultLocale(java.util.Locale)), [setDoubleBuffered](http://docs.google.com/javax/swing/JComponent.html#setDoubleBuffered(boolean)), [setEnabled](http://docs.google.com/javax/swing/JComponent.html#setEnabled(boolean)), [setFocusTraversalKeys](http://docs.google.com/javax/swing/JComponent.html#setFocusTraversalKeys(int,%20java.util.Set)), [setFont](http://docs.google.com/javax/swing/JComponent.html#setFont(java.awt.Font)), [setForeground](http://docs.google.com/javax/swing/JComponent.html#setForeground(java.awt.Color)), [setInheritsPopupMenu](http://docs.google.com/javax/swing/JComponent.html#setInheritsPopupMenu(boolean)), [setInputMap](http://docs.google.com/javax/swing/JComponent.html#setInputMap(int,%20javax.swing.InputMap)), [setInputVerifier](http://docs.google.com/javax/swing/JComponent.html#setInputVerifier(javax.swing.InputVerifier)), [setMaximumSize](http://docs.google.com/javax/swing/JComponent.html#setMaximumSize(java.awt.Dimension)), [setMinimumSize](http://docs.google.com/javax/swing/JComponent.html#setMinimumSize(java.awt.Dimension)), [setNextFocusableComponent](http://docs.google.com/javax/swing/JComponent.html#setNextFocusableComponent(java.awt.Component)), [setOpaque](http://docs.google.com/javax/swing/JComponent.html#setOpaque(boolean)), [setPreferredSize](http://docs.google.com/javax/swing/JComponent.html#setPreferredSize(java.awt.Dimension)), [setRequestFocusEnabled](http://docs.google.com/javax/swing/JComponent.html#setRequestFocusEnabled(boolean)), [setToolTipText](http://docs.google.com/javax/swing/JComponent.html#setToolTipText(java.lang.String)), [setTransferHandler](http://docs.google.com/javax/swing/JComponent.html#setTransferHandler(javax.swing.TransferHandler)), [setUI](http://docs.google.com/javax/swing/JComponent.html#setUI(javax.swing.plaf.ComponentUI)), [setVerifyInputWhenFocusTarget](http://docs.google.com/javax/swing/JComponent.html#setVerifyInputWhenFocusTarget(boolean)), [setVisible](http://docs.google.com/javax/swing/JComponent.html#setVisible(boolean)), [unregisterKeyboardAction](http://docs.google.com/javax/swing/JComponent.html#unregisterKeyboardAction(javax.swing.KeyStroke)), [update](http://docs.google.com/javax/swing/JComponent.html#update(java.awt.Graphics)), [updateUI](http://docs.google.com/javax/swing/JComponent.html#updateUI()) |

| **Methods inherited from class java.awt.**[**Container**](http://docs.google.com/java/awt/Container.html) |
| --- |
| [add](http://docs.google.com/java/awt/Container.html#add(java.awt.Component)), [add](http://docs.google.com/java/awt/Container.html#add(java.awt.Component,%20int)), [add](http://docs.google.com/java/awt/Container.html#add(java.awt.Component,%20java.lang.Object)), [add](http://docs.google.com/java/awt/Container.html#add(java.awt.Component,%20java.lang.Object,%20int)), [add](http://docs.google.com/java/awt/Container.html#add(java.lang.String,%20java.awt.Component)), [addContainerListener](http://docs.google.com/java/awt/Container.html#addContainerListener(java.awt.event.ContainerListener)), [addPropertyChangeListener](http://docs.google.com/java/awt/Container.html#addPropertyChangeListener(java.beans.PropertyChangeListener)), [addPropertyChangeListener](http://docs.google.com/java/awt/Container.html#addPropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener)), [applyComponentOrientation](http://docs.google.com/java/awt/Container.html#applyComponentOrientation(java.awt.ComponentOrientation)), [areFocusTraversalKeysSet](http://docs.google.com/java/awt/Container.html#areFocusTraversalKeysSet(int)), [countComponents](http://docs.google.com/java/awt/Container.html#countComponents()), [deliverEvent](http://docs.google.com/java/awt/Container.html#deliverEvent(java.awt.Event)), [doLayout](http://docs.google.com/java/awt/Container.html#doLayout()), [findComponentAt](http://docs.google.com/java/awt/Container.html#findComponentAt(int,%20int)), [findComponentAt](http://docs.google.com/java/awt/Container.html#findComponentAt(java.awt.Point)), [getComponent](http://docs.google.com/java/awt/Container.html#getComponent(int)), [getComponentAt](http://docs.google.com/java/awt/Container.html#getComponentAt(int,%20int)), [getComponentAt](http://docs.google.com/java/awt/Container.html#getComponentAt(java.awt.Point)), [getComponentCount](http://docs.google.com/java/awt/Container.html#getComponentCount()), [getComponents](http://docs.google.com/java/awt/Container.html#getComponents()), [getComponentZOrder](http://docs.google.com/java/awt/Container.html#getComponentZOrder(java.awt.Component)), [getContainerListeners](http://docs.google.com/java/awt/Container.html#getContainerListeners()), [getFocusTraversalKeys](http://docs.google.com/java/awt/Container.html#getFocusTraversalKeys(int)), [getFocusTraversalPolicy](http://docs.google.com/java/awt/Container.html#getFocusTraversalPolicy()), [getLayout](http://docs.google.com/java/awt/Container.html#getLayout()), [getMousePosition](http://docs.google.com/java/awt/Container.html#getMousePosition(boolean)), [insets](http://docs.google.com/java/awt/Container.html#insets()), [invalidate](http://docs.google.com/java/awt/Container.html#invalidate()), [isAncestorOf](http://docs.google.com/java/awt/Container.html#isAncestorOf(java.awt.Component)), [isFocusCycleRoot](http://docs.google.com/java/awt/Container.html#isFocusCycleRoot()), [isFocusCycleRoot](http://docs.google.com/java/awt/Container.html#isFocusCycleRoot(java.awt.Container)), [isFocusTraversalPolicyProvider](http://docs.google.com/java/awt/Container.html#isFocusTraversalPolicyProvider()), [isFocusTraversalPolicySet](http://docs.google.com/java/awt/Container.html#isFocusTraversalPolicySet()), [layout](http://docs.google.com/java/awt/Container.html#layout()), [list](http://docs.google.com/java/awt/Container.html#list(java.io.PrintStream,%20int)), [list](http://docs.google.com/java/awt/Container.html#list(java.io.PrintWriter,%20int)), [locate](http://docs.google.com/java/awt/Container.html#locate(int,%20int)), [minimumSize](http://docs.google.com/java/awt/Container.html#minimumSize()), [paintComponents](http://docs.google.com/java/awt/Container.html#paintComponents(java.awt.Graphics)), [preferredSize](http://docs.google.com/java/awt/Container.html#preferredSize()), [printComponents](http://docs.google.com/java/awt/Container.html#printComponents(java.awt.Graphics)), [processContainerEvent](http://docs.google.com/java/awt/Container.html#processContainerEvent(java.awt.event.ContainerEvent)), [processEvent](http://docs.google.com/java/awt/Container.html#processEvent(java.awt.AWTEvent)), [remove](http://docs.google.com/java/awt/Container.html#remove(java.awt.Component)), [removeContainerListener](http://docs.google.com/java/awt/Container.html#removeContainerListener(java.awt.event.ContainerListener)), [setComponentZOrder](http://docs.google.com/java/awt/Container.html#setComponentZOrder(java.awt.Component,%20int)), [setFocusCycleRoot](http://docs.google.com/java/awt/Container.html#setFocusCycleRoot(boolean)), [setFocusTraversalPolicy](http://docs.google.com/java/awt/Container.html#setFocusTraversalPolicy(java.awt.FocusTraversalPolicy)), [setFocusTraversalPolicyProvider](http://docs.google.com/java/awt/Container.html#setFocusTraversalPolicyProvider(boolean)), [setLayout](http://docs.google.com/java/awt/Container.html#setLayout(java.awt.LayoutManager)), [transferFocusBackward](http://docs.google.com/java/awt/Container.html#transferFocusBackward()), [transferFocusDownCycle](http://docs.google.com/java/awt/Container.html#transferFocusDownCycle()), [validate](http://docs.google.com/java/awt/Container.html#validate()), [validateTree](http://docs.google.com/java/awt/Container.html#validateTree()) |

| **Methods inherited from class java.awt.**[**Component**](http://docs.google.com/java/awt/Component.html) |
| --- |
| [action](http://docs.google.com/java/awt/Component.html#action(java.awt.Event,%20java.lang.Object)), [add](http://docs.google.com/java/awt/Component.html#add(java.awt.PopupMenu)), [addComponentListener](http://docs.google.com/java/awt/Component.html#addComponentListener(java.awt.event.ComponentListener)), [addFocusListener](http://docs.google.com/java/awt/Component.html#addFocusListener(java.awt.event.FocusListener)), [addHierarchyBoundsListener](http://docs.google.com/java/awt/Component.html#addHierarchyBoundsListener(java.awt.event.HierarchyBoundsListener)), [addHierarchyListener](http://docs.google.com/java/awt/Component.html#addHierarchyListener(java.awt.event.HierarchyListener)), [addInputMethodListener](http://docs.google.com/java/awt/Component.html#addInputMethodListener(java.awt.event.InputMethodListener)), [addKeyListener](http://docs.google.com/java/awt/Component.html#addKeyListener(java.awt.event.KeyListener)), [addMouseListener](http://docs.google.com/java/awt/Component.html#addMouseListener(java.awt.event.MouseListener)), [addMouseMotionListener](http://docs.google.com/java/awt/Component.html#addMouseMotionListener(java.awt.event.MouseMotionListener)), [addMouseWheelListener](http://docs.google.com/java/awt/Component.html#addMouseWheelListener(java.awt.event.MouseWheelListener)), [bounds](http://docs.google.com/java/awt/Component.html#bounds()), [checkImage](http://docs.google.com/java/awt/Component.html#checkImage(java.awt.Image,%20java.awt.image.ImageObserver)), [checkImage](http://docs.google.com/java/awt/Component.html#checkImage(java.awt.Image,%20int,%20int,%20java.awt.image.ImageObserver)), [coalesceEvents](http://docs.google.com/java/awt/Component.html#coalesceEvents(java.awt.AWTEvent,%20java.awt.AWTEvent)), [contains](http://docs.google.com/java/awt/Component.html#contains(java.awt.Point)), [createImage](http://docs.google.com/java/awt/Component.html#createImage(java.awt.image.ImageProducer)), [createImage](http://docs.google.com/java/awt/Component.html#createImage(int,%20int)), [createVolatileImage](http://docs.google.com/java/awt/Component.html#createVolatileImage(int,%20int)), [createVolatileImage](http://docs.google.com/java/awt/Component.html#createVolatileImage(int,%20int,%20java.awt.ImageCapabilities)), [disableEvents](http://docs.google.com/java/awt/Component.html#disableEvents(long)), [dispatchEvent](http://docs.google.com/java/awt/Component.html#dispatchEvent(java.awt.AWTEvent)), [enable](http://docs.google.com/java/awt/Component.html#enable(boolean)), [enableEvents](http://docs.google.com/java/awt/Component.html#enableEvents(long)), [enableInputMethods](http://docs.google.com/java/awt/Component.html#enableInputMethods(boolean)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20byte,%20byte)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20double,%20double)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20float,%20float)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20long,%20long)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object)), [firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20short,%20short)), [getBackground](http://docs.google.com/java/awt/Component.html#getBackground()), [getBounds](http://docs.google.com/java/awt/Component.html#getBounds()), [getColorModel](http://docs.google.com/java/awt/Component.html#getColorModel()), [getComponentListeners](http://docs.google.com/java/awt/Component.html#getComponentListeners()), [getComponentOrientation](http://docs.google.com/java/awt/Component.html#getComponentOrientation()), [getCursor](http://docs.google.com/java/awt/Component.html#getCursor()), [getDropTarget](http://docs.google.com/java/awt/Component.html#getDropTarget()), [getFocusCycleRootAncestor](http://docs.google.com/java/awt/Component.html#getFocusCycleRootAncestor()), [getFocusListeners](http://docs.google.com/java/awt/Component.html#getFocusListeners()), [getFocusTraversalKeysEnabled](http://docs.google.com/java/awt/Component.html#getFocusTraversalKeysEnabled()), [getFont](http://docs.google.com/java/awt/Component.html#getFont()), [getForeground](http://docs.google.com/java/awt/Component.html#getForeground()), [getGraphicsConfiguration](http://docs.google.com/java/awt/Component.html#getGraphicsConfiguration()), [getHierarchyBoundsListeners](http://docs.google.com/java/awt/Component.html#getHierarchyBoundsListeners()), [getHierarchyListeners](http://docs.google.com/java/awt/Component.html#getHierarchyListeners()), [getIgnoreRepaint](http://docs.google.com/java/awt/Component.html#getIgnoreRepaint()), [getInputContext](http://docs.google.com/java/awt/Component.html#getInputContext()), [getInputMethodListeners](http://docs.google.com/java/awt/Component.html#getInputMethodListeners()), [getInputMethodRequests](http://docs.google.com/java/awt/Component.html#getInputMethodRequests()), [getKeyListeners](http://docs.google.com/java/awt/Component.html#getKeyListeners()), [getLocale](http://docs.google.com/java/awt/Component.html#getLocale()), [getLocation](http://docs.google.com/java/awt/Component.html#getLocation()), [getLocationOnScreen](http://docs.google.com/java/awt/Component.html#getLocationOnScreen()), [getMouseListeners](http://docs.google.com/java/awt/Component.html#getMouseListeners()), [getMouseMotionListeners](http://docs.google.com/java/awt/Component.html#getMouseMotionListeners()), [getMousePosition](http://docs.google.com/java/awt/Component.html#getMousePosition()), [getMouseWheelListeners](http://docs.google.com/java/awt/Component.html#getMouseWheelListeners()), [getName](http://docs.google.com/java/awt/Component.html#getName()), [getParent](http://docs.google.com/java/awt/Component.html#getParent()), [getPeer](http://docs.google.com/java/awt/Component.html#getPeer()), [getPropertyChangeListeners](http://docs.google.com/java/awt/Component.html#getPropertyChangeListeners()), [getPropertyChangeListeners](http://docs.google.com/java/awt/Component.html#getPropertyChangeListeners(java.lang.String)), [getSize](http://docs.google.com/java/awt/Component.html#getSize()), [getToolkit](http://docs.google.com/java/awt/Component.html#getToolkit()), [getTreeLock](http://docs.google.com/java/awt/Component.html#getTreeLock()), [gotFocus](http://docs.google.com/java/awt/Component.html#gotFocus(java.awt.Event,%20java.lang.Object)), [handleEvent](http://docs.google.com/java/awt/Component.html#handleEvent(java.awt.Event)), [hasFocus](http://docs.google.com/java/awt/Component.html#hasFocus()), [hide](http://docs.google.com/java/awt/Component.html#hide()), [imageUpdate](http://docs.google.com/java/awt/Component.html#imageUpdate(java.awt.Image,%20int,%20int,%20int,%20int,%20int)), [inside](http://docs.google.com/java/awt/Component.html#inside(int,%20int)), [isBackgroundSet](http://docs.google.com/java/awt/Component.html#isBackgroundSet()), [isCursorSet](http://docs.google.com/java/awt/Component.html#isCursorSet()), [isDisplayable](http://docs.google.com/java/awt/Component.html#isDisplayable()), [isEnabled](http://docs.google.com/java/awt/Component.html#isEnabled()), [isFocusable](http://docs.google.com/java/awt/Component.html#isFocusable()), [isFocusOwner](http://docs.google.com/java/awt/Component.html#isFocusOwner()), [isFocusTraversable](http://docs.google.com/java/awt/Component.html#isFocusTraversable()), [isFontSet](http://docs.google.com/java/awt/Component.html#isFontSet()), [isForegroundSet](http://docs.google.com/java/awt/Component.html#isForegroundSet()), [isLightweight](http://docs.google.com/java/awt/Component.html#isLightweight()), [isMaximumSizeSet](http://docs.google.com/java/awt/Component.html#isMaximumSizeSet()), [isMinimumSizeSet](http://docs.google.com/java/awt/Component.html#isMinimumSizeSet()), [isPreferredSizeSet](http://docs.google.com/java/awt/Component.html#isPreferredSizeSet()), [isShowing](http://docs.google.com/java/awt/Component.html#isShowing()), [isValid](http://docs.google.com/java/awt/Component.html#isValid()), [isVisible](http://docs.google.com/java/awt/Component.html#isVisible()), [keyDown](http://docs.google.com/java/awt/Component.html#keyDown(java.awt.Event,%20int)), [keyUp](http://docs.google.com/java/awt/Component.html#keyUp(java.awt.Event,%20int)), [list](http://docs.google.com/java/awt/Component.html#list()), [list](http://docs.google.com/java/awt/Component.html#list(java.io.PrintStream)), [list](http://docs.google.com/java/awt/Component.html#list(java.io.PrintWriter)), [location](http://docs.google.com/java/awt/Component.html#location()), [lostFocus](http://docs.google.com/java/awt/Component.html#lostFocus(java.awt.Event,%20java.lang.Object)), [mouseDown](http://docs.google.com/java/awt/Component.html#mouseDown(java.awt.Event,%20int,%20int)), [mouseDrag](http://docs.google.com/java/awt/Component.html#mouseDrag(java.awt.Event,%20int,%20int)), [mouseEnter](http://docs.google.com/java/awt/Component.html#mouseEnter(java.awt.Event,%20int,%20int)), [mouseExit](http://docs.google.com/java/awt/Component.html#mouseExit(java.awt.Event,%20int,%20int)), [mouseMove](http://docs.google.com/java/awt/Component.html#mouseMove(java.awt.Event,%20int,%20int)), [mouseUp](http://docs.google.com/java/awt/Component.html#mouseUp(java.awt.Event,%20int,%20int)), [move](http://docs.google.com/java/awt/Component.html#move(int,%20int)), [nextFocus](http://docs.google.com/java/awt/Component.html#nextFocus()), [paintAll](http://docs.google.com/java/awt/Component.html#paintAll(java.awt.Graphics)), [postEvent](http://docs.google.com/java/awt/Component.html#postEvent(java.awt.Event)), [prepareImage](http://docs.google.com/java/awt/Component.html#prepareImage(java.awt.Image,%20java.awt.image.ImageObserver)), [prepareImage](http://docs.google.com/java/awt/Component.html#prepareImage(java.awt.Image,%20int,%20int,%20java.awt.image.ImageObserver)), [processComponentEvent](http://docs.google.com/java/awt/Component.html#processComponentEvent(java.awt.event.ComponentEvent)), [processFocusEvent](http://docs.google.com/java/awt/Component.html#processFocusEvent(java.awt.event.FocusEvent)), [processHierarchyBoundsEvent](http://docs.google.com/java/awt/Component.html#processHierarchyBoundsEvent(java.awt.event.HierarchyEvent)), [processHierarchyEvent](http://docs.google.com/java/awt/Component.html#processHierarchyEvent(java.awt.event.HierarchyEvent)), [processInputMethodEvent](http://docs.google.com/java/awt/Component.html#processInputMethodEvent(java.awt.event.InputMethodEvent)), [processMouseWheelEvent](http://docs.google.com/java/awt/Component.html#processMouseWheelEvent(java.awt.event.MouseWheelEvent)), [remove](http://docs.google.com/java/awt/Component.html#remove(java.awt.MenuComponent)), [removeComponentListener](http://docs.google.com/java/awt/Component.html#removeComponentListener(java.awt.event.ComponentListener)), [removeFocusListener](http://docs.google.com/java/awt/Component.html#removeFocusListener(java.awt.event.FocusListener)), [removeHierarchyBoundsListener](http://docs.google.com/java/awt/Component.html#removeHierarchyBoundsListener(java.awt.event.HierarchyBoundsListener)), [removeHierarchyListener](http://docs.google.com/java/awt/Component.html#removeHierarchyListener(java.awt.event.HierarchyListener)), [removeInputMethodListener](http://docs.google.com/java/awt/Component.html#removeInputMethodListener(java.awt.event.InputMethodListener)), [removeKeyListener](http://docs.google.com/java/awt/Component.html#removeKeyListener(java.awt.event.KeyListener)), [removeMouseListener](http://docs.google.com/java/awt/Component.html#removeMouseListener(java.awt.event.MouseListener)), [removeMouseMotionListener](http://docs.google.com/java/awt/Component.html#removeMouseMotionListener(java.awt.event.MouseMotionListener)), [removeMouseWheelListener](http://docs.google.com/java/awt/Component.html#removeMouseWheelListener(java.awt.event.MouseWheelListener)), [removePropertyChangeListener](http://docs.google.com/java/awt/Component.html#removePropertyChangeListener(java.beans.PropertyChangeListener)), [removePropertyChangeListener](http://docs.google.com/java/awt/Component.html#removePropertyChangeListener(java.lang.String,%20java.beans.PropertyChangeListener)), [repaint](http://docs.google.com/java/awt/Component.html#repaint()), [repaint](http://docs.google.com/java/awt/Component.html#repaint(int,%20int,%20int,%20int)), [repaint](http://docs.google.com/java/awt/Component.html#repaint(long)), [resize](http://docs.google.com/java/awt/Component.html#resize(java.awt.Dimension)), [resize](http://docs.google.com/java/awt/Component.html#resize(int,%20int)), [setBounds](http://docs.google.com/java/awt/Component.html#setBounds(int,%20int,%20int,%20int)), [setBounds](http://docs.google.com/java/awt/Component.html#setBounds(java.awt.Rectangle)), [setComponentOrientation](http://docs.google.com/java/awt/Component.html#setComponentOrientation(java.awt.ComponentOrientation)), [setCursor](http://docs.google.com/java/awt/Component.html#setCursor(java.awt.Cursor)), [setDropTarget](http://docs.google.com/java/awt/Component.html#setDropTarget(java.awt.dnd.DropTarget)), [setFocusable](http://docs.google.com/java/awt/Component.html#setFocusable(boolean)), [setFocusTraversalKeysEnabled](http://docs.google.com/java/awt/Component.html#setFocusTraversalKeysEnabled(boolean)), [setIgnoreRepaint](http://docs.google.com/java/awt/Component.html#setIgnoreRepaint(boolean)), [setLocale](http://docs.google.com/java/awt/Component.html#setLocale(java.util.Locale)), [setLocation](http://docs.google.com/java/awt/Component.html#setLocation(int,%20int)), [setLocation](http://docs.google.com/java/awt/Component.html#setLocation(java.awt.Point)), [setName](http://docs.google.com/java/awt/Component.html#setName(java.lang.String)), [setSize](http://docs.google.com/java/awt/Component.html#setSize(java.awt.Dimension)), [setSize](http://docs.google.com/java/awt/Component.html#setSize(int,%20int)), [show](http://docs.google.com/java/awt/Component.html#show()), [show](http://docs.google.com/java/awt/Component.html#show(boolean)), [size](http://docs.google.com/java/awt/Component.html#size()), [toString](http://docs.google.com/java/awt/Component.html#toString()), [transferFocus](http://docs.google.com/java/awt/Component.html#transferFocus()), [transferFocusUpCycle](http://docs.google.com/java/awt/Component.html#transferFocusUpCycle()) |

| **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()), [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** |
| --- |

### DEFAULT\_LAYER

public static final [Integer](http://docs.google.com/java/lang/Integer.html) **DEFAULT\_LAYER**

Convenience object defining the Default layer. Equivalent to new Integer(0).

### PALETTE\_LAYER

public static final [Integer](http://docs.google.com/java/lang/Integer.html) **PALETTE\_LAYER**

Convenience object defining the Palette layer. Equivalent to new Integer(100).

### MODAL\_LAYER

public static final [Integer](http://docs.google.com/java/lang/Integer.html) **MODAL\_LAYER**

Convenience object defining the Modal layer. Equivalent to new Integer(200).

### POPUP\_LAYER

public static final [Integer](http://docs.google.com/java/lang/Integer.html) **POPUP\_LAYER**

Convenience object defining the Popup layer. Equivalent to new Integer(300).

### DRAG\_LAYER

public static final [Integer](http://docs.google.com/java/lang/Integer.html) **DRAG\_LAYER**

Convenience object defining the Drag layer. Equivalent to new Integer(400).

### FRAME\_CONTENT\_LAYER

public static final [Integer](http://docs.google.com/java/lang/Integer.html) **FRAME\_CONTENT\_LAYER**

Convenience object defining the Frame Content layer. This layer is normally only use to positon the contentPane and menuBar components of JFrame. Equivalent to new Integer(-30000).

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

### LAYER\_PROPERTY

public static final [String](http://docs.google.com/java/lang/String.html) **LAYER\_PROPERTY**

Bound property

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#javax.swing.JLayeredPane.LAYER_PROPERTY)

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

### JLayeredPane

public **JLayeredPane**()

Create a new JLayeredPane

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

### addImpl

protected void **addImpl**([Component](http://docs.google.com/java/awt/Component.html) comp,  
 [Object](http://docs.google.com/java/lang/Object.html) constraints,  
 int index)

**Description copied from class:** [**Container**](http://docs.google.com/java/awt/Container.html#addImpl(java.awt.Component,%20java.lang.Object,%20int)) Adds the specified component to this container at the specified index. This method also notifies the layout manager to add the component to this container's layout using the specified constraints object via the addLayoutComponent method.

The constraints are defined by the particular layout manager being used. For example, the BorderLayout class defines five constraints: BorderLayout.NORTH, BorderLayout.SOUTH, BorderLayout.EAST, BorderLayout.WEST, and BorderLayout.CENTER.

The GridBagLayout class requires a GridBagConstraints object. Failure to pass the correct type of constraints object results in an IllegalArgumentException.

If the current layout manager implements LayoutManager2, then [LayoutManager2.addLayoutComponent(Component,Object)](http://docs.google.com/java/awt/LayoutManager2.html#addLayoutComponent(java.awt.Component,%20java.lang.Object)) is invoked on it. If the current layout manager does not implement LayoutManager2, and constraints is a String, then [LayoutManager.addLayoutComponent(String,Component)](http://docs.google.com/java/awt/LayoutManager.html#addLayoutComponent(java.lang.String,%20java.awt.Component)) is invoked on it.

If the component is not an ancestor of this container and has a non-null parent, it is removed from its current parent before it is added to this container.

This is the method to override if a program needs to track every add request to a container as all other add methods defer to this one. An overriding method should usually include a call to the superclass's version of the method:

super.addImpl(comp, constraints, index)

**Overrides:**[addImpl](http://docs.google.com/java/awt/Container.html#addImpl(java.awt.Component,%20java.lang.Object,%20int)) in class [Container](http://docs.google.com/java/awt/Container.html) **Parameters:**comp - the component to be addedconstraints - an object expressing layout constraints for this componentindex - the position in the container's list at which to insert the component, where -1 means append to the end**See Also:**[Container.add(Component)](http://docs.google.com/java/awt/Container.html#add(java.awt.Component)), [Container.add(Component, int)](http://docs.google.com/java/awt/Container.html#add(java.awt.Component,%20int)), [Container.add(Component, java.lang.Object)](http://docs.google.com/java/awt/Container.html#add(java.awt.Component,%20java.lang.Object)), [LayoutManager](http://docs.google.com/java/awt/LayoutManager.html), [LayoutManager2](http://docs.google.com/java/awt/LayoutManager2.html)

### remove

public void **remove**(int index)

Remove the indexed component from this pane. This is the absolute index, ignoring layers.

**Overrides:**[remove](http://docs.google.com/java/awt/Container.html#remove(int)) in class [Container](http://docs.google.com/java/awt/Container.html) **Parameters:**index - an int specifying the component to remove**See Also:**[getIndexOf(java.awt.Component)](http://docs.google.com/javax/swing/JLayeredPane.html#getIndexOf(java.awt.Component))

### removeAll

public void **removeAll**()

Removes all the components from this container.

**Overrides:**[removeAll](http://docs.google.com/java/awt/Container.html#removeAll()) in class [Container](http://docs.google.com/java/awt/Container.html) **Since:** 1.5 **See Also:**[Container.add(java.awt.Component)](http://docs.google.com/java/awt/Container.html#add(java.awt.Component)), [Container.remove(int)](http://docs.google.com/java/awt/Container.html#remove(int))

### isOptimizedDrawingEnabled

public boolean **isOptimizedDrawingEnabled**()

Returns false if components in the pane can overlap, which makes optimized drawing impossible. Otherwise, returns true.

**Overrides:**[isOptimizedDrawingEnabled](http://docs.google.com/javax/swing/JComponent.html#isOptimizedDrawingEnabled()) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Returns:**false if components can overlap, else true**See Also:**[JComponent.isOptimizedDrawingEnabled()](http://docs.google.com/javax/swing/JComponent.html#isOptimizedDrawingEnabled())

### putLayer

public static void **putLayer**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c,  
 int layer)

Sets the layer property on a JComponent. This method does not cause any side effects like setLayer() (painting, add/remove, etc). Normally you should use the instance method setLayer(), in order to get the desired side-effects (like repainting).

**Parameters:**c - the JComponent to movelayer - an int specifying the layer to move it to**See Also:**[setLayer(java.awt.Component, int)](http://docs.google.com/javax/swing/JLayeredPane.html#setLayer(java.awt.Component,%20int))

### getLayer

public static int **getLayer**([JComponent](http://docs.google.com/javax/swing/JComponent.html) c)

Gets the layer property for a JComponent, it does not cause any side effects like setLayer(). (painting, add/remove, etc) Normally you should use the instance method getLayer().

**Parameters:**c - the JComponent to check **Returns:**an int specifying the component's layer

### getLayeredPaneAbove

public static [JLayeredPane](http://docs.google.com/javax/swing/JLayeredPane.html) **getLayeredPaneAbove**([Component](http://docs.google.com/java/awt/Component.html) c)

Convenience method that returns the first JLayeredPane which contains the specified component. Note that all JFrames have a JLayeredPane at their root, so any component in a JFrame will have a JLayeredPane parent.

**Parameters:**c - the Component to check **Returns:**the JLayeredPane that contains the component, or null if no JLayeredPane is found in the component hierarchy**See Also:**[JFrame](http://docs.google.com/javax/swing/JFrame.html), [JRootPane](http://docs.google.com/javax/swing/JRootPane.html)

### setLayer

public void **setLayer**([Component](http://docs.google.com/java/awt/Component.html) c,  
 int layer)

Sets the layer attribute on the specified component, making it the bottommost component in that layer. Should be called before adding to parent.

**Parameters:**c - the Component to set the layer forlayer - an int specifying the layer to set, where lower numbers are closer to the bottom

### setLayer

public void **setLayer**([Component](http://docs.google.com/java/awt/Component.html) c,  
 int layer,  
 int position)

Sets the layer attribute for the specified component and also sets its position within that layer.

**Parameters:**c - the Component to set the layer forlayer - an int specifying the layer to set, where lower numbers are closer to the bottomposition - an int specifying the position within the layer, where 0 is the topmost position and -1 is the bottommost position

### getLayer

public int **getLayer**([Component](http://docs.google.com/java/awt/Component.html) c)

Returns the layer attribute for the specified Component.

**Parameters:**c - the Component to check **Returns:**an int specifying the component's current layer

### getIndexOf

public int **getIndexOf**([Component](http://docs.google.com/java/awt/Component.html) c)

Returns the index of the specified Component. This is the absolute index, ignoring layers. Index numbers, like position numbers, have the topmost component at index zero. Larger numbers are closer to the bottom.

**Parameters:**c - the Component to check **Returns:**an int specifying the component's index

### moveToFront

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

Moves the component to the top of the components in its current layer (position 0).

**Parameters:**c - the Component to move**See Also:**[setPosition(Component, int)](http://docs.google.com/javax/swing/JLayeredPane.html#setPosition(java.awt.Component,%20int))

### moveToBack

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

Moves the component to the bottom of the components in its current layer (position -1).

**Parameters:**c - the Component to move**See Also:**[setPosition(Component, int)](http://docs.google.com/javax/swing/JLayeredPane.html#setPosition(java.awt.Component,%20int))

### setPosition

public void **setPosition**([Component](http://docs.google.com/java/awt/Component.html) c,  
 int position)

Moves the component to position within its current layer, where 0 is the topmost position within the layer and -1 is the bottommost position.

**Note:** Position numbering is defined by java.awt.Container, and is the opposite of layer numbering. Lower position numbers are closer to the top (0 is topmost), and higher position numbers are closer to the bottom.

**Parameters:**c - the Component to moveposition - an int in the range -1..N-1, where N is the number of components in the component's current layer

### getPosition

public int **getPosition**([Component](http://docs.google.com/java/awt/Component.html) c)

Get the relative position of the component within its layer.

**Parameters:**c - the Component to check **Returns:**an int giving the component's position, where 0 is the topmost position and the highest index value = the count count of components at that layer, minus 1**See Also:**[getComponentCountInLayer(int)](http://docs.google.com/javax/swing/JLayeredPane.html#getComponentCountInLayer(int))

### highestLayer

public int **highestLayer**()

Returns the highest layer value from all current children. Returns 0 if there are no children.

**Returns:**an int indicating the layer of the topmost component in the pane, or zero if there are no children

### lowestLayer

public int **lowestLayer**()

Returns the lowest layer value from all current children. Returns 0 if there are no children.

**Returns:**an int indicating the layer of the bottommost component in the pane, or zero if there are no children

### getComponentCountInLayer

public int **getComponentCountInLayer**(int layer)

Returns the number of children currently in the specified layer.

**Parameters:**layer - an int specifying the layer to check **Returns:**an int specifying the number of components in that layer

### getComponentsInLayer

public [Component](http://docs.google.com/java/awt/Component.html)[] **getComponentsInLayer**(int layer)

Returns an array of the components in the specified layer.

**Parameters:**layer - an int specifying the layer to check **Returns:**an array of Components contained in that layer

### paint

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

Paints this JLayeredPane within the specified graphics context.

**Overrides:**[paint](http://docs.google.com/javax/swing/JComponent.html#paint(java.awt.Graphics)) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Parameters:**g - the Graphics context within which to paint**See Also:**[JComponent.paintComponent(java.awt.Graphics)](http://docs.google.com/javax/swing/JComponent.html#paintComponent(java.awt.Graphics)), [JComponent.paintBorder(java.awt.Graphics)](http://docs.google.com/javax/swing/JComponent.html#paintBorder(java.awt.Graphics)), [JComponent.paintChildren(java.awt.Graphics)](http://docs.google.com/javax/swing/JComponent.html#paintChildren(java.awt.Graphics)), [JComponent.getComponentGraphics(java.awt.Graphics)](http://docs.google.com/javax/swing/JComponent.html#getComponentGraphics(java.awt.Graphics)), [JComponent.repaint(long, int, int, int, int)](http://docs.google.com/javax/swing/JComponent.html#repaint(long,%20int,%20int,%20int,%20int))

### getComponentToLayer

protected [Hashtable](http://docs.google.com/java/util/Hashtable.html)<[Component](http://docs.google.com/java/awt/Component.html),[Integer](http://docs.google.com/java/lang/Integer.html)> **getComponentToLayer**()

Returns the hashtable that maps components to layers.

**Returns:**the Hashtable used to map components to their layers

### getObjectForLayer

protected [Integer](http://docs.google.com/java/lang/Integer.html) **getObjectForLayer**(int layer)

Returns the Integer object associated with a specified layer.

**Parameters:**layer - an int specifying the layer **Returns:**an Integer object for that layer

### insertIndexForLayer

protected int **insertIndexForLayer**(int layer,  
 int position)

Primitive method that determines the proper location to insert a new child based on layer and position requests.

**Parameters:**layer - an int specifying the layerposition - an int specifying the position within the layer **Returns:**an int giving the (absolute) insertion-index**See Also:**[getIndexOf(java.awt.Component)](http://docs.google.com/javax/swing/JLayeredPane.html#getIndexOf(java.awt.Component))

### paramString

protected [String](http://docs.google.com/java/lang/String.html) **paramString**()

Returns a string representation of this JLayeredPane. This method is intended to be used only for debugging purposes, and the content and format of the returned string may vary between implementations. The returned string may be empty but may not be null.

**Overrides:**[paramString](http://docs.google.com/javax/swing/JComponent.html#paramString()) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Returns:**a string representation of this JLayeredPane.

### getAccessibleContext

public [AccessibleContext](http://docs.google.com/javax/accessibility/AccessibleContext.html) **getAccessibleContext**()

Gets the AccessibleContext associated with this JLayeredPane. For layered panes, the AccessibleContext takes the form of an AccessibleJLayeredPane. A new AccessibleJLayeredPane instance is created if necessary.

**Specified by:**[getAccessibleContext](http://docs.google.com/javax/accessibility/Accessible.html#getAccessibleContext()) in interface [Accessible](http://docs.google.com/javax/accessibility/Accessible.html)**Overrides:**[getAccessibleContext](http://docs.google.com/javax/swing/JComponent.html#getAccessibleContext()) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Returns:**an AccessibleJLayeredPane that serves as the AccessibleContext of this JLayeredPane

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/JLayeredPane.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/JLabel.AccessibleJLabel.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/JLayeredPane.AccessibleJLayeredPane.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/JLayeredPane.html)    [**NO FRAMES**](http://docs.google.com/JLayeredPane.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#1t3h5sf) | [CONSTR](#3rdcrjn) | [METHOD](#26in1rg) | DETAIL: [FIELD](#2jxsxqh) | [CONSTR](#2bn6wsx) | [METHOD](#3as4poj) |

[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).