| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DefaultTreeCellRenderer.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/tree/DefaultTreeCellEditor.EditorContainer.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/tree/DefaultTreeCellRenderer.html)    [**NO FRAMES**](http://docs.google.com/DefaultTreeCellRenderer.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#tyjcwt) | [FIELD](#2s8eyo1) | [CONSTR](#1ksv4uv) | [METHOD](#44sinio) | DETAIL: [FIELD](#2xcytpi) | [CONSTR](#23ckvvd) | [METHOD](#32hioqz) |

## **javax.swing.tree**

Class DefaultTreeCellRenderer

[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.JLabel](http://docs.google.com/javax/swing/JLabel.html)  
 **javax.swing.tree.DefaultTreeCellRenderer**

**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), [SwingConstants](http://docs.google.com/javax/swing/SwingConstants.html), [TreeCellRenderer](http://docs.google.com/javax/swing/tree/TreeCellRenderer.html)

public class **DefaultTreeCellRenderer**extends [JLabel](http://docs.google.com/javax/swing/JLabel.html)implements [TreeCellRenderer](http://docs.google.com/javax/swing/tree/TreeCellRenderer.html)

Displays an entry in a tree. DefaultTreeCellRenderer is not opaque and unless you subclass paint you should not change this. See [How to Use Trees](http://java.sun.com/docs/books/tutorial/uiswing/components/tree.html) in *The Java Tutorial* for examples of customizing node display using this class.

**Implementation Note:** This class overrides invalidate, validate, revalidate, repaint, and firePropertyChange solely to improve performance. If not overridden, these frequently called methods would execute code paths that are unnecessary for the default tree cell renderer. If you write your own renderer, take care to weigh the benefits and drawbacks of overriding these methods.

**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** | |
| --- | --- |

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

| **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** | |
| --- | --- |
| protected  [Color](http://docs.google.com/java/awt/Color.html) | [**backgroundNonSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#backgroundNonSelectionColor)            Color to use for the background when the node isn't selected. |
| protected  [Color](http://docs.google.com/java/awt/Color.html) | [**backgroundSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#backgroundSelectionColor)            Color to use for the background when a node is selected. |
| protected  [Color](http://docs.google.com/java/awt/Color.html) | [**borderSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#borderSelectionColor)            Color to use for the focus indicator when the node has focus. |
| protected  [Icon](http://docs.google.com/javax/swing/Icon.html) | [**closedIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#closedIcon)            Icon used to show non-leaf nodes that aren't expanded. |
| protected  boolean | [**hasFocus**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#hasFocus)            True if has focus. |
| protected  [Icon](http://docs.google.com/javax/swing/Icon.html) | [**leafIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#leafIcon)            Icon used to show leaf nodes. |
| protected  [Icon](http://docs.google.com/javax/swing/Icon.html) | [**openIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#openIcon)            Icon used to show non-leaf nodes that are expanded. |
| protected  boolean | [**selected**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#selected)            Is the value currently selected. |
| protected  [Color](http://docs.google.com/java/awt/Color.html) | [**textNonSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#textNonSelectionColor)            Color to use for the foreground for non-selected nodes. |
| protected  [Color](http://docs.google.com/java/awt/Color.html) | [**textSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#textSelectionColor)            Color to use for the foreground for selected nodes. |

| **Fields inherited from class javax.swing.**[**JLabel**](http://docs.google.com/javax/swing/JLabel.html) |
| --- |
| [labelFor](http://docs.google.com/javax/swing/JLabel.html#labelFor) |

| **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 javax.swing.**[**SwingConstants**](http://docs.google.com/javax/swing/SwingConstants.html) |
| --- |
| [BOTTOM](http://docs.google.com/javax/swing/SwingConstants.html#BOTTOM), [CENTER](http://docs.google.com/javax/swing/SwingConstants.html#CENTER), [EAST](http://docs.google.com/javax/swing/SwingConstants.html#EAST), [HORIZONTAL](http://docs.google.com/javax/swing/SwingConstants.html#HORIZONTAL), [LEADING](http://docs.google.com/javax/swing/SwingConstants.html#LEADING), [LEFT](http://docs.google.com/javax/swing/SwingConstants.html#LEFT), [NEXT](http://docs.google.com/javax/swing/SwingConstants.html#NEXT), [NORTH](http://docs.google.com/javax/swing/SwingConstants.html#NORTH), [NORTH\_EAST](http://docs.google.com/javax/swing/SwingConstants.html#NORTH_EAST), [NORTH\_WEST](http://docs.google.com/javax/swing/SwingConstants.html#NORTH_WEST), [PREVIOUS](http://docs.google.com/javax/swing/SwingConstants.html#PREVIOUS), [RIGHT](http://docs.google.com/javax/swing/SwingConstants.html#RIGHT), [SOUTH](http://docs.google.com/javax/swing/SwingConstants.html#SOUTH), [SOUTH\_EAST](http://docs.google.com/javax/swing/SwingConstants.html#SOUTH_EAST), [SOUTH\_WEST](http://docs.google.com/javax/swing/SwingConstants.html#SOUTH_WEST), [TOP](http://docs.google.com/javax/swing/SwingConstants.html#TOP), [TRAILING](http://docs.google.com/javax/swing/SwingConstants.html#TRAILING), [VERTICAL](http://docs.google.com/javax/swing/SwingConstants.html#VERTICAL), [WEST](http://docs.google.com/javax/swing/SwingConstants.html#WEST) |

| **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** | |
| --- | --- |
| [**DefaultTreeCellRenderer**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#DefaultTreeCellRenderer())()            Returns a new instance of DefaultTreeCellRenderer. |

| **Method Summary** | |
| --- | --- |
| void | [**firePropertyChange**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#firePropertyChange(java.lang.String,%20boolean,%20boolean))([String](http://docs.google.com/java/lang/String.html) propertyName, boolean oldValue, boolean newValue)            Overridden for performance reasons. |
| void | [**firePropertyChange**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#firePropertyChange(java.lang.String,%20byte,%20byte))([String](http://docs.google.com/java/lang/String.html) propertyName, byte oldValue, byte newValue)            Overridden for performance reasons. |
| void | [**firePropertyChange**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#firePropertyChange(java.lang.String,%20char,%20char))([String](http://docs.google.com/java/lang/String.html) propertyName, char oldValue, char newValue)            Overridden for performance reasons. |
| void | [**firePropertyChange**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#firePropertyChange(java.lang.String,%20double,%20double))([String](http://docs.google.com/java/lang/String.html) propertyName, double oldValue, double newValue)            Overridden for performance reasons. |
| void | [**firePropertyChange**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#firePropertyChange(java.lang.String,%20float,%20float))([String](http://docs.google.com/java/lang/String.html) propertyName, float oldValue, float newValue)            Overridden for performance reasons. |
| void | [**firePropertyChange**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#firePropertyChange(java.lang.String,%20int,%20int))([String](http://docs.google.com/java/lang/String.html) propertyName, int oldValue, int newValue)            Overridden for performance reasons. |
| void | [**firePropertyChange**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#firePropertyChange(java.lang.String,%20long,%20long))([String](http://docs.google.com/java/lang/String.html) propertyName, long oldValue, long newValue)            Overridden for performance reasons. |
| protected  void | [**firePropertyChange**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object))([String](http://docs.google.com/java/lang/String.html) propertyName, [Object](http://docs.google.com/java/lang/Object.html) oldValue, [Object](http://docs.google.com/java/lang/Object.html) newValue)            Overridden for performance reasons. |
| void | [**firePropertyChange**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#firePropertyChange(java.lang.String,%20short,%20short))([String](http://docs.google.com/java/lang/String.html) propertyName, short oldValue, short newValue)            Overridden for performance reasons. |
| [Color](http://docs.google.com/java/awt/Color.html) | [**getBackgroundNonSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getBackgroundNonSelectionColor())()            Returns the background color to be used for non selected nodes. |
| [Color](http://docs.google.com/java/awt/Color.html) | [**getBackgroundSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getBackgroundSelectionColor())()            Returns the color to use for the background if node is selected. |
| [Color](http://docs.google.com/java/awt/Color.html) | [**getBorderSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getBorderSelectionColor())()            Returns the color the border is drawn. |
| [Icon](http://docs.google.com/javax/swing/Icon.html) | [**getClosedIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getClosedIcon())()            Returns the icon used to represent non-leaf nodes that are not expanded. |
| [Icon](http://docs.google.com/javax/swing/Icon.html) | [**getDefaultClosedIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getDefaultClosedIcon())()            Returns the default icon, for the current laf, that is used to represent non-leaf nodes that are not expanded. |
| [Icon](http://docs.google.com/javax/swing/Icon.html) | [**getDefaultLeafIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getDefaultLeafIcon())()            Returns the default icon, for the current laf, that is used to represent leaf nodes. |
| [Icon](http://docs.google.com/javax/swing/Icon.html) | [**getDefaultOpenIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getDefaultOpenIcon())()            Returns the default icon, for the current laf, that is used to represent non-leaf nodes that are expanded. |
| [Font](http://docs.google.com/java/awt/Font.html) | [**getFont**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getFont())()            Gets the font of this component. |
| [Icon](http://docs.google.com/javax/swing/Icon.html) | [**getLeafIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getLeafIcon())()            Returns the icon used to represent leaf nodes. |
| [Icon](http://docs.google.com/javax/swing/Icon.html) | [**getOpenIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getOpenIcon())()            Returns the icon used to represent non-leaf nodes that are expanded. |
| [Dimension](http://docs.google.com/java/awt/Dimension.html) | [**getPreferredSize**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getPreferredSize())()            Overrides JComponent.getPreferredSize to return slightly wider preferred size value. |
| [Color](http://docs.google.com/java/awt/Color.html) | [**getTextNonSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getTextNonSelectionColor())()            Returns the color the text is drawn with when the node isn't selected. |
| [Color](http://docs.google.com/java/awt/Color.html) | [**getTextSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getTextSelectionColor())()            Returns the color the text is drawn with when the node is selected. |
| [Component](http://docs.google.com/java/awt/Component.html) | [**getTreeCellRendererComponent**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#getTreeCellRendererComponent(javax.swing.JTree,%20java.lang.Object,%20boolean,%20boolean,%20boolean,%20int,%20boolean))([JTree](http://docs.google.com/javax/swing/JTree.html) tree, [Object](http://docs.google.com/java/lang/Object.html) value, boolean sel, boolean expanded, boolean leaf, int row, boolean hasFocus)            Configures the renderer based on the passed in components. |
| void | [**invalidate**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#invalidate())()            Overridden for performance reasons. |
| void | [**paint**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#paint(java.awt.Graphics))([Graphics](http://docs.google.com/java/awt/Graphics.html) g)            Paints the value. |
| void | [**repaint**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#repaint())()            Overridden for performance reasons. |
| void | [**repaint**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#repaint(long,%20int,%20int,%20int,%20int))(long tm, int x, int y, int width, int height)            Overridden for performance reasons. |
| void | [**repaint**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#repaint(java.awt.Rectangle))([Rectangle](http://docs.google.com/java/awt/Rectangle.html) r)            Overridden for performance reasons. |
| void | [**revalidate**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#revalidate())()            Overridden for performance reasons. |
| void | [**setBackground**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#setBackground(java.awt.Color))([Color](http://docs.google.com/java/awt/Color.html) color)            Subclassed to map ColorUIResources to null. |
| void | [**setBackgroundNonSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#setBackgroundNonSelectionColor(java.awt.Color))([Color](http://docs.google.com/java/awt/Color.html) newColor)            Sets the background color to be used for non selected nodes. |
| void | [**setBackgroundSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#setBackgroundSelectionColor(java.awt.Color))([Color](http://docs.google.com/java/awt/Color.html) newColor)            Sets the color to use for the background if node is selected. |
| void | [**setBorderSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#setBorderSelectionColor(java.awt.Color))([Color](http://docs.google.com/java/awt/Color.html) newColor)            Sets the color to use for the border. |
| void | [**setClosedIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#setClosedIcon(javax.swing.Icon))([Icon](http://docs.google.com/javax/swing/Icon.html) newIcon)            Sets the icon used to represent non-leaf nodes that are not expanded. |
| void | [**setFont**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#setFont(java.awt.Font))([Font](http://docs.google.com/java/awt/Font.html) font)            Subclassed to map FontUIResources to null. |
| void | [**setLeafIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#setLeafIcon(javax.swing.Icon))([Icon](http://docs.google.com/javax/swing/Icon.html) newIcon)            Sets the icon used to represent leaf nodes. |
| void | [**setOpenIcon**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#setOpenIcon(javax.swing.Icon))([Icon](http://docs.google.com/javax/swing/Icon.html) newIcon)            Sets the icon used to represent non-leaf nodes that are expanded. |
| void | [**setTextNonSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#setTextNonSelectionColor(java.awt.Color))([Color](http://docs.google.com/java/awt/Color.html) newColor)            Sets the color the text is drawn with when the node isn't selected. |
| void | [**setTextSelectionColor**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#setTextSelectionColor(java.awt.Color))([Color](http://docs.google.com/java/awt/Color.html) newColor)            Sets the color the text is drawn with when the node is selected. |
| void | [**validate**](http://docs.google.com/javax/swing/tree/DefaultTreeCellRenderer.html#validate())()            Overridden for performance reasons. |

| **Methods inherited from class javax.swing.**[**JLabel**](http://docs.google.com/javax/swing/JLabel.html) |
| --- |
| [checkHorizontalKey](http://docs.google.com/javax/swing/JLabel.html#checkHorizontalKey(int,%20java.lang.String)), [checkVerticalKey](http://docs.google.com/javax/swing/JLabel.html#checkVerticalKey(int,%20java.lang.String)), [getAccessibleContext](http://docs.google.com/javax/swing/JLabel.html#getAccessibleContext()), [getDisabledIcon](http://docs.google.com/javax/swing/JLabel.html#getDisabledIcon()), [getDisplayedMnemonic](http://docs.google.com/javax/swing/JLabel.html#getDisplayedMnemonic()), [getDisplayedMnemonicIndex](http://docs.google.com/javax/swing/JLabel.html#getDisplayedMnemonicIndex()), [getHorizontalAlignment](http://docs.google.com/javax/swing/JLabel.html#getHorizontalAlignment()), [getHorizontalTextPosition](http://docs.google.com/javax/swing/JLabel.html#getHorizontalTextPosition()), [getIcon](http://docs.google.com/javax/swing/JLabel.html#getIcon()), [getIconTextGap](http://docs.google.com/javax/swing/JLabel.html#getIconTextGap()), [getLabelFor](http://docs.google.com/javax/swing/JLabel.html#getLabelFor()), [getText](http://docs.google.com/javax/swing/JLabel.html#getText()), [getUI](http://docs.google.com/javax/swing/JLabel.html#getUI()), [getUIClassID](http://docs.google.com/javax/swing/JLabel.html#getUIClassID()), [getVerticalAlignment](http://docs.google.com/javax/swing/JLabel.html#getVerticalAlignment()), [getVerticalTextPosition](http://docs.google.com/javax/swing/JLabel.html#getVerticalTextPosition()), [imageUpdate](http://docs.google.com/javax/swing/JLabel.html#imageUpdate(java.awt.Image,%20int,%20int,%20int,%20int,%20int)), [paramString](http://docs.google.com/javax/swing/JLabel.html#paramString()), [setDisabledIcon](http://docs.google.com/javax/swing/JLabel.html#setDisabledIcon(javax.swing.Icon)), [setDisplayedMnemonic](http://docs.google.com/javax/swing/JLabel.html#setDisplayedMnemonic(char)), [setDisplayedMnemonic](http://docs.google.com/javax/swing/JLabel.html#setDisplayedMnemonic(int)), [setDisplayedMnemonicIndex](http://docs.google.com/javax/swing/JLabel.html#setDisplayedMnemonicIndex(int)), [setHorizontalAlignment](http://docs.google.com/javax/swing/JLabel.html#setHorizontalAlignment(int)), [setHorizontalTextPosition](http://docs.google.com/javax/swing/JLabel.html#setHorizontalTextPosition(int)), [setIcon](http://docs.google.com/javax/swing/JLabel.html#setIcon(javax.swing.Icon)), [setIconTextGap](http://docs.google.com/javax/swing/JLabel.html#setIconTextGap(int)), [setLabelFor](http://docs.google.com/javax/swing/JLabel.html#setLabelFor(java.awt.Component)), [setText](http://docs.google.com/javax/swing/JLabel.html#setText(java.lang.String)), [setUI](http://docs.google.com/javax/swing/JLabel.html#setUI(javax.swing.plaf.LabelUI)), [setVerticalAlignment](http://docs.google.com/javax/swing/JLabel.html#setVerticalAlignment(int)), [setVerticalTextPosition](http://docs.google.com/javax/swing/JLabel.html#setVerticalTextPosition(int)), [updateUI](http://docs.google.com/javax/swing/JLabel.html#updateUI()) |

| **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()), [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)), [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()), [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()), [isOptimizedDrawingEnabled](http://docs.google.com/javax/swing/JComponent.html#isOptimizedDrawingEnabled()), [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)), [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)), [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)), [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)), [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)) |

| **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)), [addImpl](http://docs.google.com/java/awt/Container.html#addImpl(java.awt.Component,%20java.lang.Object,%20int)), [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()), [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)), [remove](http://docs.google.com/java/awt/Container.html#remove(int)), [removeAll](http://docs.google.com/java/awt/Container.html#removeAll()), [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()), [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)), [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()), [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()), [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(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** |
| --- |

### selected

protected boolean **selected**

Is the value currently selected.

### hasFocus

protected boolean **hasFocus**

True if has focus.

### closedIcon

protected transient [Icon](http://docs.google.com/javax/swing/Icon.html) **closedIcon**

Icon used to show non-leaf nodes that aren't expanded.

### leafIcon

protected transient [Icon](http://docs.google.com/javax/swing/Icon.html) **leafIcon**

Icon used to show leaf nodes.

### openIcon

protected transient [Icon](http://docs.google.com/javax/swing/Icon.html) **openIcon**

Icon used to show non-leaf nodes that are expanded.

### textSelectionColor

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

Color to use for the foreground for selected nodes.

### textNonSelectionColor

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

Color to use for the foreground for non-selected nodes.

### backgroundSelectionColor

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

Color to use for the background when a node is selected.

### backgroundNonSelectionColor

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

Color to use for the background when the node isn't selected.

### borderSelectionColor

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

Color to use for the focus indicator when the node has focus.

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

### DefaultTreeCellRenderer

public **DefaultTreeCellRenderer**()

Returns a new instance of DefaultTreeCellRenderer. Alignment is set to left aligned. Icons and text color are determined from the UIManager.

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

### getDefaultOpenIcon

public [Icon](http://docs.google.com/javax/swing/Icon.html) **getDefaultOpenIcon**()

Returns the default icon, for the current laf, that is used to represent non-leaf nodes that are expanded.

### getDefaultClosedIcon

public [Icon](http://docs.google.com/javax/swing/Icon.html) **getDefaultClosedIcon**()

Returns the default icon, for the current laf, that is used to represent non-leaf nodes that are not expanded.

### getDefaultLeafIcon

public [Icon](http://docs.google.com/javax/swing/Icon.html) **getDefaultLeafIcon**()

Returns the default icon, for the current laf, that is used to represent leaf nodes.

### setOpenIcon

public void **setOpenIcon**([Icon](http://docs.google.com/javax/swing/Icon.html) newIcon)

Sets the icon used to represent non-leaf nodes that are expanded.

### getOpenIcon

public [Icon](http://docs.google.com/javax/swing/Icon.html) **getOpenIcon**()

Returns the icon used to represent non-leaf nodes that are expanded.

### setClosedIcon

public void **setClosedIcon**([Icon](http://docs.google.com/javax/swing/Icon.html) newIcon)

Sets the icon used to represent non-leaf nodes that are not expanded.

### getClosedIcon

public [Icon](http://docs.google.com/javax/swing/Icon.html) **getClosedIcon**()

Returns the icon used to represent non-leaf nodes that are not expanded.

### setLeafIcon

public void **setLeafIcon**([Icon](http://docs.google.com/javax/swing/Icon.html) newIcon)

Sets the icon used to represent leaf nodes.

### getLeafIcon

public [Icon](http://docs.google.com/javax/swing/Icon.html) **getLeafIcon**()

Returns the icon used to represent leaf nodes.

### setTextSelectionColor

public void **setTextSelectionColor**([Color](http://docs.google.com/java/awt/Color.html) newColor)

Sets the color the text is drawn with when the node is selected.

### getTextSelectionColor

public [Color](http://docs.google.com/java/awt/Color.html) **getTextSelectionColor**()

Returns the color the text is drawn with when the node is selected.

### setTextNonSelectionColor

public void **setTextNonSelectionColor**([Color](http://docs.google.com/java/awt/Color.html) newColor)

Sets the color the text is drawn with when the node isn't selected.

### getTextNonSelectionColor

public [Color](http://docs.google.com/java/awt/Color.html) **getTextNonSelectionColor**()

Returns the color the text is drawn with when the node isn't selected.

### setBackgroundSelectionColor

public void **setBackgroundSelectionColor**([Color](http://docs.google.com/java/awt/Color.html) newColor)

Sets the color to use for the background if node is selected.

### getBackgroundSelectionColor

public [Color](http://docs.google.com/java/awt/Color.html) **getBackgroundSelectionColor**()

Returns the color to use for the background if node is selected.

### setBackgroundNonSelectionColor

public void **setBackgroundNonSelectionColor**([Color](http://docs.google.com/java/awt/Color.html) newColor)

Sets the background color to be used for non selected nodes.

### getBackgroundNonSelectionColor

public [Color](http://docs.google.com/java/awt/Color.html) **getBackgroundNonSelectionColor**()

Returns the background color to be used for non selected nodes.

### setBorderSelectionColor

public void **setBorderSelectionColor**([Color](http://docs.google.com/java/awt/Color.html) newColor)

Sets the color to use for the border.

### getBorderSelectionColor

public [Color](http://docs.google.com/java/awt/Color.html) **getBorderSelectionColor**()

Returns the color the border is drawn.

### setFont

public void **setFont**([Font](http://docs.google.com/java/awt/Font.html) font)

Subclassed to map FontUIResources to null. If font is null, or a FontUIResource, this has the effect of letting the font of the JTree show through. On the other hand, if font is non-null, and not a FontUIResource, the font becomes font.

**Overrides:**[setFont](http://docs.google.com/javax/swing/JComponent.html#setFont(java.awt.Font)) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Parameters:**font - the desired Font for this component**See Also:**[Component.getFont()](http://docs.google.com/java/awt/Component.html#getFont())

### getFont

public [Font](http://docs.google.com/java/awt/Font.html) **getFont**()

Gets the font of this component.

**Specified by:**[getFont](http://docs.google.com/java/awt/MenuContainer.html#getFont()) in interface [MenuContainer](http://docs.google.com/java/awt/MenuContainer.html)**Overrides:**[getFont](http://docs.google.com/java/awt/Component.html#getFont()) in class [Component](http://docs.google.com/java/awt/Component.html) **Returns:**this component's font; if a font has not been set for this component, the font of its parent is returned**See Also:**[Component.setFont(java.awt.Font)](http://docs.google.com/java/awt/Component.html#setFont(java.awt.Font))

### setBackground

public void **setBackground**([Color](http://docs.google.com/java/awt/Color.html) color)

Subclassed to map ColorUIResources to null. If color is null, or a ColorUIResource, this has the effect of letting the background color of the JTree show through. On the other hand, if color is non-null, and not a ColorUIResource, the background becomes color.

**Overrides:**[setBackground](http://docs.google.com/javax/swing/JComponent.html#setBackground(java.awt.Color)) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Parameters:**color - the desired background Color**See Also:**[Component.getBackground()](http://docs.google.com/java/awt/Component.html#getBackground()), [JComponent.setOpaque(boolean)](http://docs.google.com/javax/swing/JComponent.html#setOpaque(boolean))

### getTreeCellRendererComponent

public [Component](http://docs.google.com/java/awt/Component.html) **getTreeCellRendererComponent**([JTree](http://docs.google.com/javax/swing/JTree.html) tree,  
 [Object](http://docs.google.com/java/lang/Object.html) value,  
 boolean sel,  
 boolean expanded,  
 boolean leaf,  
 int row,  
 boolean hasFocus)

Configures the renderer based on the passed in components. The value is set from messaging the tree with convertValueToText, which ultimately invokes toString on value. The foreground color is set based on the selection and the icon is set based on the leaf and expanded parameters.

**Specified by:**[getTreeCellRendererComponent](http://docs.google.com/javax/swing/tree/TreeCellRenderer.html#getTreeCellRendererComponent(javax.swing.JTree,%20java.lang.Object,%20boolean,%20boolean,%20boolean,%20int,%20boolean)) in interface [TreeCellRenderer](http://docs.google.com/javax/swing/tree/TreeCellRenderer.html) **Returns:**the Component that the renderer uses to draw the value

### paint

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

Paints the value. The background is filled based on selected.

**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 in 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))

### getPreferredSize

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

Overrides JComponent.getPreferredSize to return slightly wider preferred size value.

**Overrides:**[getPreferredSize](http://docs.google.com/javax/swing/JComponent.html#getPreferredSize()) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Returns:**the value of the preferredSize property**See Also:**[JComponent.setPreferredSize(java.awt.Dimension)](http://docs.google.com/javax/swing/JComponent.html#setPreferredSize(java.awt.Dimension)), [ComponentUI](http://docs.google.com/javax/swing/plaf/ComponentUI.html)

### validate

public void **validate**()

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[validate](http://docs.google.com/java/awt/Container.html#validate()) in class [Container](http://docs.google.com/java/awt/Container.html) **See Also:**[Container.add(java.awt.Component)](http://docs.google.com/java/awt/Container.html#add(java.awt.Component)), [Component.invalidate()](http://docs.google.com/java/awt/Component.html#invalidate()), [JComponent.revalidate()](http://docs.google.com/javax/swing/JComponent.html#revalidate()), [Container.validateTree()](http://docs.google.com/java/awt/Container.html#validateTree())

### invalidate

public void **invalidate**()

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[invalidate](http://docs.google.com/java/awt/Container.html#invalidate()) in class [Container](http://docs.google.com/java/awt/Container.html) **Since:** 1.5 **See Also:**[Container.validate()](http://docs.google.com/java/awt/Container.html#validate()), [Container.layout()](http://docs.google.com/java/awt/Container.html#layout()), [LayoutManager](http://docs.google.com/java/awt/LayoutManager.html), [LayoutManager2.invalidateLayout(Container)](http://docs.google.com/java/awt/LayoutManager2.html#invalidateLayout(java.awt.Container))

### revalidate

public void **revalidate**()

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[revalidate](http://docs.google.com/javax/swing/JComponent.html#revalidate()) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **See Also:**[Component.invalidate()](http://docs.google.com/java/awt/Component.html#invalidate()), [Container.validate()](http://docs.google.com/java/awt/Container.html#validate()), [JComponent.isValidateRoot()](http://docs.google.com/javax/swing/JComponent.html#isValidateRoot()), [RepaintManager.addInvalidComponent(javax.swing.JComponent)](http://docs.google.com/javax/swing/RepaintManager.html#addInvalidComponent(javax.swing.JComponent))

### repaint

public void **repaint**(long tm,  
 int x,  
 int y,  
 int width,  
 int height)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[repaint](http://docs.google.com/javax/swing/JComponent.html#repaint(long,%20int,%20int,%20int,%20int)) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Parameters:**tm - this parameter is not usedx - the x value of the dirty regiony - the y value of the dirty regionwidth - the width of the dirty regionheight - the height of the dirty region**See Also:**[Component.isShowing()](http://docs.google.com/java/awt/Component.html#isShowing()), [RepaintManager.addDirtyRegion(javax.swing.JComponent, int, int, int, int)](http://docs.google.com/javax/swing/RepaintManager.html#addDirtyRegion(javax.swing.JComponent,%20int,%20int,%20int,%20int))

### repaint

public void **repaint**([Rectangle](http://docs.google.com/java/awt/Rectangle.html) r)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[repaint](http://docs.google.com/javax/swing/JComponent.html#repaint(java.awt.Rectangle)) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Parameters:**r - a Rectangle containing the dirty region**See Also:**[Component.isShowing()](http://docs.google.com/java/awt/Component.html#isShowing()), [RepaintManager.addDirtyRegion(javax.swing.JComponent, int, int, int, int)](http://docs.google.com/javax/swing/RepaintManager.html#addDirtyRegion(javax.swing.JComponent,%20int,%20int,%20int,%20int))

### repaint

public void **repaint**()

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[repaint](http://docs.google.com/java/awt/Component.html#repaint()) in class [Component](http://docs.google.com/java/awt/Component.html) **Since:** 1.5 **See Also:**[Component.update(Graphics)](http://docs.google.com/java/awt/Component.html#update(java.awt.Graphics))

### firePropertyChange

protected void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 [Object](http://docs.google.com/java/lang/Object.html) oldValue,  
 [Object](http://docs.google.com/java/lang/Object.html) newValue)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object)) in class [Component](http://docs.google.com/java/awt/Component.html) **Parameters:**propertyName - the property whose value has changedoldValue - the property's previous valuenewValue - the property's new value

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 byte oldValue,  
 byte newValue)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20byte,%20byte)) in class [Component](http://docs.google.com/java/awt/Component.html) **Parameters:**propertyName - the programmatic name of the property that was changedoldValue - the old value of the property (as a byte)newValue - the new value of the property (as a byte)**See Also:**[Component.firePropertyChange(java.lang.String, java.lang.Object, java.lang.Object)](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object))

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 char oldValue,  
 char newValue)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[firePropertyChange](http://docs.google.com/javax/swing/JComponent.html#firePropertyChange(java.lang.String,%20char,%20char)) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Parameters:**propertyName - the programmatic name of the property that was changedoldValue - the old value of the property (as a char)newValue - the new value of the property (as a char)**See Also:**[Component.firePropertyChange(java.lang.String, java.lang.Object, java.lang.Object)](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object))

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 short oldValue,  
 short newValue)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20short,%20short)) in class [Component](http://docs.google.com/java/awt/Component.html) **Parameters:**propertyName - the programmatic name of the property that was changedoldValue - the old value of the property (as a short)newValue - the old value of the property (as a short)**See Also:**[Component.firePropertyChange(java.lang.String, java.lang.Object, java.lang.Object)](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object))

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 int oldValue,  
 int newValue)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[firePropertyChange](http://docs.google.com/javax/swing/JComponent.html#firePropertyChange(java.lang.String,%20int,%20int)) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Parameters:**propertyName - the property whose value has changedoldValue - the property's previous valuenewValue - the property's new value

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 long oldValue,  
 long newValue)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20long,%20long)) in class [Component](http://docs.google.com/java/awt/Component.html) **Parameters:**propertyName - the programmatic name of the property that was changedoldValue - the old value of the property (as a long)newValue - the new value of the property (as a long)**See Also:**[Component.firePropertyChange(java.lang.String, java.lang.Object, java.lang.Object)](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object))

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 float oldValue,  
 float newValue)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20float,%20float)) in class [Component](http://docs.google.com/java/awt/Component.html) **Parameters:**propertyName - the programmatic name of the property that was changedoldValue - the old value of the property (as a float)newValue - the new value of the property (as a float)**See Also:**[Component.firePropertyChange(java.lang.String, java.lang.Object, java.lang.Object)](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object))

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 double oldValue,  
 double newValue)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[firePropertyChange](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20double,%20double)) in class [Component](http://docs.google.com/java/awt/Component.html) **Parameters:**propertyName - the programmatic name of the property that was changedoldValue - the old value of the property (as a double)newValue - the new value of the property (as a double)**See Also:**[Component.firePropertyChange(java.lang.String, java.lang.Object, java.lang.Object)](http://docs.google.com/java/awt/Component.html#firePropertyChange(java.lang.String,%20java.lang.Object,%20java.lang.Object))

### firePropertyChange

public void **firePropertyChange**([String](http://docs.google.com/java/lang/String.html) propertyName,  
 boolean oldValue,  
 boolean newValue)

Overridden for performance reasons. See the [Implementation Note](#3znysh7) for more information.

**Overrides:**[firePropertyChange](http://docs.google.com/javax/swing/JComponent.html#firePropertyChange(java.lang.String,%20boolean,%20boolean)) in class [JComponent](http://docs.google.com/javax/swing/JComponent.html) **Parameters:**propertyName - the property whose value has changedoldValue - the property's previous valuenewValue - the property's new value

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DefaultTreeCellRenderer.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/tree/DefaultTreeCellEditor.EditorContainer.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/tree/DefaultTreeCellRenderer.html)    [**NO FRAMES**](http://docs.google.com/DefaultTreeCellRenderer.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#tyjcwt) | [FIELD](#2s8eyo1) | [CONSTR](#1ksv4uv) | [METHOD](#44sinio) | DETAIL: [FIELD](#2xcytpi) | [CONSTR](#23ckvvd) | [METHOD](#32hioqz) |

[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).