| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/JFormattedTextField.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/JFileChooser.AccessibleJFileChooser.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/JFormattedTextField.html)    [**NO FRAMES**](http://docs.google.com/JFormattedTextField.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2s8eyo1) | [CONSTR](#44sinio) | [METHOD](#2jxsxqh) | DETAIL: [FIELD](#3whwml4) | [CONSTR](#49x2ik5) | [METHOD](#1hmsyys) |

## **javax.swing**

Class JFormattedTextField

[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.text.JTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.html)  
 [javax.swing.JTextField](http://docs.google.com/javax/swing/JTextField.html)  
 **javax.swing.JFormattedTextField**

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

public class **JFormattedTextField**extends [JTextField](http://docs.google.com/javax/swing/JTextField.html)

JFormattedTextField extends JTextField adding support for formatting arbitrary values, as well as retrieving a particular object once the user has edited the text. The following illustrates configuring a JFormattedTextField to edit dates:

JFormattedTextField ftf = new JFormattedTextField();  
 ftf.setValue(new Date());

Once a JFormattedTextField has been created, you can listen for editing changes by way of adding a PropertyChangeListener and listening for PropertyChangeEvents with the property name value.

JFormattedTextField allows configuring what action should be taken when focus is lost. The possible configurations are:

| Value | Description |
| --- | --- |
| JFormattedTextField.REVERT | Revert the display to match that of getValue, possibly losing the current edit. |
| JFormattedTextField.COMMIT | Commits the current value. If the value being edited isn't considered a legal value by the AbstractFormatter that is, a ParseException is thrown, then the value will not change, and then edited value will persist. |
| JFormattedTextField.COMMIT\_OR\_REVERT | Similar to COMMIT, but if the value isn't legal, behave like REVERT. |
| JFormattedTextField.PERSIST | Do nothing, don't obtain a new AbstractFormatter, and don't update the value. |

The default is JFormattedTextField.COMMIT\_OR\_REVERT, refer to [setFocusLostBehavior(int)](http://docs.google.com/javax/swing/JFormattedTextField.html#setFocusLostBehavior(int)) for more information on this.

JFormattedTextField allows the focus to leave, even if the currently edited value is invalid. To lock the focus down while the JFormattedTextField is an invalid edit state you can attach an InputVerifier. The following code snippet shows a potential implementation of such an InputVerifier:

public class FormattedTextFieldVerifier extends InputVerifier {  
 public boolean verify(JComponent input) {  
 if (input instanceof JFormattedTextField) {  
 JFormattedTextField ftf = (JFormattedTextField)input;  
 AbstractFormatter formatter = ftf.getFormatter();  
 if (formatter != null) {  
 String text = ftf.getText();  
 try {  
 formatter.stringToValue(text);  
 return true;  
 } catch (ParseException pe) {  
 return false;  
 }  
 }  
 }  
 return true;  
 }  
 public boolean shouldYieldFocus(JComponent input) {  
 return verify(input);  
 }  
 }

Alternatively, you could invoke commitEdit, which would also commit the value.

JFormattedTextField does not do the formatting it self, rather formatting is done through an instance of JFormattedTextField.AbstractFormatter which is obtained from an instance of JFormattedTextField.AbstractFormatterFactory. Instances of JFormattedTextField.AbstractFormatter are notified when they become active by way of the install method, at which point the JFormattedTextField.AbstractFormatter can install whatever it needs to, typically a DocumentFilter. Similarly when JFormattedTextField no longer needs the AbstractFormatter, it will invoke uninstall.

JFormattedTextField typically queries the AbstractFormatterFactory for an AbstractFormat when it gains or loses focus. Although this can change based on the focus lost policy. If the focus lost policy is JFormattedTextField.PERSIST and the JFormattedTextField has been edited, the AbstractFormatterFactory will not be queried until the value has been commited. Similarly if the focus lost policy is JFormattedTextField.COMMIT and an exception is thrown from stringToValue, the AbstractFormatterFactory will not be querired when focus is lost or gained.

JFormattedTextField.AbstractFormatter is also responsible for determining when values are commited to the JFormattedTextField. Some JFormattedTextField.AbstractFormatters will make new values available on every edit, and others will never commit the value. You can force the current value to be obtained from the current JFormattedTextField.AbstractFormatter by way of invoking commitEdit. commitEdit will be invoked whenever return is pressed in the JFormattedTextField.

If an AbstractFormatterFactory has not been explicitly set, one will be set based on the Class of the value type after setValue has been invoked (assuming value is non-null). For example, in the following code an appropriate AbstractFormatterFactory and AbstractFormatter will be created to handle formatting of numbers:

JFormattedTextField tf = new JFormattedTextField();  
 tf.setValue(new Number(100));

**Warning:** As the AbstractFormatter will typically install a DocumentFilter on the Document, and a NavigationFilter on the JFormattedTextField you should not install your own. If you do, you are likely to see odd behavior in that the editing policy of the AbstractFormatter will not be enforced.

**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).

**Since:** 1.4

| **Nested Class Summary** | |
| --- | --- |
| static class | [**JFormattedTextField.AbstractFormatter**](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html)            Instances of AbstractFormatter are used by JFormattedTextField to handle the conversion both from an Object to a String, and back from a String to an Object. |
| static class | [**JFormattedTextField.AbstractFormatterFactory**](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatterFactory.html)            Instances of AbstractFormatterFactory are used by JFormattedTextField to obtain instances of AbstractFormatter which in turn are used to format values. |

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

| **Nested classes/interfaces inherited from class javax.swing.text.**[**JTextComponent**](http://docs.google.com/javax/swing/text/JTextComponent.html) |
| --- |
| [JTextComponent.AccessibleJTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.AccessibleJTextComponent.html), [JTextComponent.DropLocation](http://docs.google.com/javax/swing/text/JTextComponent.DropLocation.html), [JTextComponent.KeyBinding](http://docs.google.com/javax/swing/text/JTextComponent.KeyBinding.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** | |
| --- | --- |
| static int | [**COMMIT**](http://docs.google.com/javax/swing/JFormattedTextField.html#COMMIT)            Constant identifying that when focus is lost, commitEdit should be invoked. |
| static int | [**COMMIT\_OR\_REVERT**](http://docs.google.com/javax/swing/JFormattedTextField.html#COMMIT_OR_REVERT)            Constant identifying that when focus is lost, commitEdit should be invoked. |
| static int | [**PERSIST**](http://docs.google.com/javax/swing/JFormattedTextField.html#PERSIST)            Constant identifying that when focus is lost, the edited value should be left. |
| static int | [**REVERT**](http://docs.google.com/javax/swing/JFormattedTextField.html#REVERT)            Constant identifying that when focus is lost, editing value should be reverted to current value set on the JFormattedTextField. |

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

| **Fields inherited from class javax.swing.text.**[**JTextComponent**](http://docs.google.com/javax/swing/text/JTextComponent.html) |
| --- |
| [DEFAULT\_KEYMAP](http://docs.google.com/javax/swing/text/JTextComponent.html#DEFAULT_KEYMAP), [FOCUS\_ACCELERATOR\_KEY](http://docs.google.com/javax/swing/text/JTextComponent.html#FOCUS_ACCELERATOR_KEY) |

| **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** | |
| --- | --- |
| [**JFormattedTextField**](http://docs.google.com/javax/swing/JFormattedTextField.html#JFormattedTextField())()            Creates a JFormattedTextField with no AbstractFormatterFactory. |
| [**JFormattedTextField**](http://docs.google.com/javax/swing/JFormattedTextField.html#JFormattedTextField(java.text.Format))([Format](http://docs.google.com/java/text/Format.html) format)            Creates a JFormattedTextField. |
| [**JFormattedTextField**](http://docs.google.com/javax/swing/JFormattedTextField.html#JFormattedTextField(javax.swing.JFormattedTextField.AbstractFormatter))([JFormattedTextField.AbstractFormatter](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html) formatter)            Creates a JFormattedTextField with the specified AbstractFormatter. |
| [**JFormattedTextField**](http://docs.google.com/javax/swing/JFormattedTextField.html#JFormattedTextField(javax.swing.JFormattedTextField.AbstractFormatterFactory))([JFormattedTextField.AbstractFormatterFactory](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatterFactory.html) factory)            Creates a JFormattedTextField with the specified AbstractFormatterFactory. |
| [**JFormattedTextField**](http://docs.google.com/javax/swing/JFormattedTextField.html#JFormattedTextField(javax.swing.JFormattedTextField.AbstractFormatterFactory,%20java.lang.Object))([JFormattedTextField.AbstractFormatterFactory](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatterFactory.html) factory, [Object](http://docs.google.com/java/lang/Object.html) currentValue)            Creates a JFormattedTextField with the specified AbstractFormatterFactory and initial value. |
| [**JFormattedTextField**](http://docs.google.com/javax/swing/JFormattedTextField.html#JFormattedTextField(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) value)            Creates a JFormattedTextField with the specified value. |

| **Method Summary** | |
| --- | --- |
| void | [**commitEdit**](http://docs.google.com/javax/swing/JFormattedTextField.html#commitEdit())()            Forces the current value to be taken from the AbstractFormatter and set as the current value. |
| [Action](http://docs.google.com/javax/swing/Action.html)[] | [**getActions**](http://docs.google.com/javax/swing/JFormattedTextField.html#getActions())()            Fetches the command list for the editor. |
| int | [**getFocusLostBehavior**](http://docs.google.com/javax/swing/JFormattedTextField.html#getFocusLostBehavior())()            Returns the behavior when focus is lost. |
| [JFormattedTextField.AbstractFormatter](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html) | [**getFormatter**](http://docs.google.com/javax/swing/JFormattedTextField.html#getFormatter())()            Returns the AbstractFormatter that is used to format and parse the current value. |
| [JFormattedTextField.AbstractFormatterFactory](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatterFactory.html) | [**getFormatterFactory**](http://docs.google.com/javax/swing/JFormattedTextField.html#getFormatterFactory())()            Returns the current AbstractFormatterFactory. |
| [String](http://docs.google.com/java/lang/String.html) | [**getUIClassID**](http://docs.google.com/javax/swing/JFormattedTextField.html#getUIClassID())()            Gets the class ID for a UI. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getValue**](http://docs.google.com/javax/swing/JFormattedTextField.html#getValue())()            Returns the last valid value. |
| protected  void | [**invalidEdit**](http://docs.google.com/javax/swing/JFormattedTextField.html#invalidEdit())()            Invoked when the user inputs an invalid value. |
| boolean | [**isEditValid**](http://docs.google.com/javax/swing/JFormattedTextField.html#isEditValid())()            Returns true if the current value being edited is valid. |
| protected  void | [**processFocusEvent**](http://docs.google.com/javax/swing/JFormattedTextField.html#processFocusEvent(java.awt.event.FocusEvent))([FocusEvent](http://docs.google.com/java/awt/event/FocusEvent.html) e)            Processes any focus events, such as FocusEvent.FOCUS\_GAINED or FocusEvent.FOCUS\_LOST. |
| protected  void | [**processInputMethodEvent**](http://docs.google.com/javax/swing/JFormattedTextField.html#processInputMethodEvent(java.awt.event.InputMethodEvent))([InputMethodEvent](http://docs.google.com/java/awt/event/InputMethodEvent.html) e)            Processes any input method events, such as InputMethodEvent.INPUT\_METHOD\_TEXT\_CHANGED or InputMethodEvent.CARET\_POSITION\_CHANGED. |
| void | [**setDocument**](http://docs.google.com/javax/swing/JFormattedTextField.html#setDocument(javax.swing.text.Document))([Document](http://docs.google.com/javax/swing/text/Document.html) doc)            Associates the editor with a text document. |
| void | [**setFocusLostBehavior**](http://docs.google.com/javax/swing/JFormattedTextField.html#setFocusLostBehavior(int))(int behavior)            Sets the behavior when focus is lost. |
| protected  void | [**setFormatter**](http://docs.google.com/javax/swing/JFormattedTextField.html#setFormatter(javax.swing.JFormattedTextField.AbstractFormatter))([JFormattedTextField.AbstractFormatter](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html) format)            Sets the current AbstractFormatter. |
| void | [**setFormatterFactory**](http://docs.google.com/javax/swing/JFormattedTextField.html#setFormatterFactory(javax.swing.JFormattedTextField.AbstractFormatterFactory))([JFormattedTextField.AbstractFormatterFactory](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatterFactory.html) tf)            Sets the AbstractFormatterFactory. |
| void | [**setValue**](http://docs.google.com/javax/swing/JFormattedTextField.html#setValue(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) value)            Sets the value that will be formatted by an AbstractFormatter obtained from the current AbstractFormatterFactory. |

| **Methods inherited from class javax.swing.**[**JTextField**](http://docs.google.com/javax/swing/JTextField.html) |
| --- |
| [actionPropertyChanged](http://docs.google.com/javax/swing/JTextField.html#actionPropertyChanged(javax.swing.Action,%20java.lang.String)), [addActionListener](http://docs.google.com/javax/swing/JTextField.html#addActionListener(java.awt.event.ActionListener)), [configurePropertiesFromAction](http://docs.google.com/javax/swing/JTextField.html#configurePropertiesFromAction(javax.swing.Action)), [createActionPropertyChangeListener](http://docs.google.com/javax/swing/JTextField.html#createActionPropertyChangeListener(javax.swing.Action)), [createDefaultModel](http://docs.google.com/javax/swing/JTextField.html#createDefaultModel()), [fireActionPerformed](http://docs.google.com/javax/swing/JTextField.html#fireActionPerformed()), [getAccessibleContext](http://docs.google.com/javax/swing/JTextField.html#getAccessibleContext()), [getAction](http://docs.google.com/javax/swing/JTextField.html#getAction()), [getActionListeners](http://docs.google.com/javax/swing/JTextField.html#getActionListeners()), [getColumns](http://docs.google.com/javax/swing/JTextField.html#getColumns()), [getColumnWidth](http://docs.google.com/javax/swing/JTextField.html#getColumnWidth()), [getHorizontalAlignment](http://docs.google.com/javax/swing/JTextField.html#getHorizontalAlignment()), [getHorizontalVisibility](http://docs.google.com/javax/swing/JTextField.html#getHorizontalVisibility()), [getPreferredSize](http://docs.google.com/javax/swing/JTextField.html#getPreferredSize()), [getScrollOffset](http://docs.google.com/javax/swing/JTextField.html#getScrollOffset()), [isValidateRoot](http://docs.google.com/javax/swing/JTextField.html#isValidateRoot()), [paramString](http://docs.google.com/javax/swing/JTextField.html#paramString()), [postActionEvent](http://docs.google.com/javax/swing/JTextField.html#postActionEvent()), [removeActionListener](http://docs.google.com/javax/swing/JTextField.html#removeActionListener(java.awt.event.ActionListener)), [scrollRectToVisible](http://docs.google.com/javax/swing/JTextField.html#scrollRectToVisible(java.awt.Rectangle)), [setAction](http://docs.google.com/javax/swing/JTextField.html#setAction(javax.swing.Action)), [setActionCommand](http://docs.google.com/javax/swing/JTextField.html#setActionCommand(java.lang.String)), [setColumns](http://docs.google.com/javax/swing/JTextField.html#setColumns(int)), [setFont](http://docs.google.com/javax/swing/JTextField.html#setFont(java.awt.Font)), [setHorizontalAlignment](http://docs.google.com/javax/swing/JTextField.html#setHorizontalAlignment(int)), [setScrollOffset](http://docs.google.com/javax/swing/JTextField.html#setScrollOffset(int)) |

| **Methods inherited from class javax.swing.text.**[**JTextComponent**](http://docs.google.com/javax/swing/text/JTextComponent.html) |
| --- |
| [addCaretListener](http://docs.google.com/javax/swing/text/JTextComponent.html#addCaretListener(javax.swing.event.CaretListener)), [addInputMethodListener](http://docs.google.com/javax/swing/text/JTextComponent.html#addInputMethodListener(java.awt.event.InputMethodListener)), [addKeymap](http://docs.google.com/javax/swing/text/JTextComponent.html#addKeymap(java.lang.String,%20javax.swing.text.Keymap)), [copy](http://docs.google.com/javax/swing/text/JTextComponent.html#copy()), [cut](http://docs.google.com/javax/swing/text/JTextComponent.html#cut()), [fireCaretUpdate](http://docs.google.com/javax/swing/text/JTextComponent.html#fireCaretUpdate(javax.swing.event.CaretEvent)), [getCaret](http://docs.google.com/javax/swing/text/JTextComponent.html#getCaret()), [getCaretColor](http://docs.google.com/javax/swing/text/JTextComponent.html#getCaretColor()), [getCaretListeners](http://docs.google.com/javax/swing/text/JTextComponent.html#getCaretListeners()), [getCaretPosition](http://docs.google.com/javax/swing/text/JTextComponent.html#getCaretPosition()), [getDisabledTextColor](http://docs.google.com/javax/swing/text/JTextComponent.html#getDisabledTextColor()), [getDocument](http://docs.google.com/javax/swing/text/JTextComponent.html#getDocument()), [getDragEnabled](http://docs.google.com/javax/swing/text/JTextComponent.html#getDragEnabled()), [getDropLocation](http://docs.google.com/javax/swing/text/JTextComponent.html#getDropLocation()), [getDropMode](http://docs.google.com/javax/swing/text/JTextComponent.html#getDropMode()), [getFocusAccelerator](http://docs.google.com/javax/swing/text/JTextComponent.html#getFocusAccelerator()), [getHighlighter](http://docs.google.com/javax/swing/text/JTextComponent.html#getHighlighter()), [getInputMethodRequests](http://docs.google.com/javax/swing/text/JTextComponent.html#getInputMethodRequests()), [getKeymap](http://docs.google.com/javax/swing/text/JTextComponent.html#getKeymap()), [getKeymap](http://docs.google.com/javax/swing/text/JTextComponent.html#getKeymap(java.lang.String)), [getMargin](http://docs.google.com/javax/swing/text/JTextComponent.html#getMargin()), [getNavigationFilter](http://docs.google.com/javax/swing/text/JTextComponent.html#getNavigationFilter()), [getPreferredScrollableViewportSize](http://docs.google.com/javax/swing/text/JTextComponent.html#getPreferredScrollableViewportSize()), [getPrintable](http://docs.google.com/javax/swing/text/JTextComponent.html#getPrintable(java.text.MessageFormat,%20java.text.MessageFormat)), [getScrollableBlockIncrement](http://docs.google.com/javax/swing/text/JTextComponent.html#getScrollableBlockIncrement(java.awt.Rectangle,%20int,%20int)), [getScrollableTracksViewportHeight](http://docs.google.com/javax/swing/text/JTextComponent.html#getScrollableTracksViewportHeight()), [getScrollableTracksViewportWidth](http://docs.google.com/javax/swing/text/JTextComponent.html#getScrollableTracksViewportWidth()), [getScrollableUnitIncrement](http://docs.google.com/javax/swing/text/JTextComponent.html#getScrollableUnitIncrement(java.awt.Rectangle,%20int,%20int)), [getSelectedText](http://docs.google.com/javax/swing/text/JTextComponent.html#getSelectedText()), [getSelectedTextColor](http://docs.google.com/javax/swing/text/JTextComponent.html#getSelectedTextColor()), [getSelectionColor](http://docs.google.com/javax/swing/text/JTextComponent.html#getSelectionColor()), [getSelectionEnd](http://docs.google.com/javax/swing/text/JTextComponent.html#getSelectionEnd()), [getSelectionStart](http://docs.google.com/javax/swing/text/JTextComponent.html#getSelectionStart()), [getText](http://docs.google.com/javax/swing/text/JTextComponent.html#getText()), [getText](http://docs.google.com/javax/swing/text/JTextComponent.html#getText(int,%20int)), [getToolTipText](http://docs.google.com/javax/swing/text/JTextComponent.html#getToolTipText(java.awt.event.MouseEvent)), [getUI](http://docs.google.com/javax/swing/text/JTextComponent.html#getUI()), [isEditable](http://docs.google.com/javax/swing/text/JTextComponent.html#isEditable()), [loadKeymap](http://docs.google.com/javax/swing/text/JTextComponent.html#loadKeymap(javax.swing.text.Keymap,%20javax.swing.text.JTextComponent.KeyBinding%5B%5D,%20javax.swing.Action%5B%5D)), [modelToView](http://docs.google.com/javax/swing/text/JTextComponent.html#modelToView(int)), [moveCaretPosition](http://docs.google.com/javax/swing/text/JTextComponent.html#moveCaretPosition(int)), [paste](http://docs.google.com/javax/swing/text/JTextComponent.html#paste()), [print](http://docs.google.com/javax/swing/text/JTextComponent.html#print()), [print](http://docs.google.com/javax/swing/text/JTextComponent.html#print(java.text.MessageFormat,%20java.text.MessageFormat)), [print](http://docs.google.com/javax/swing/text/JTextComponent.html#print(java.text.MessageFormat,%20java.text.MessageFormat,%20boolean,%20javax.print.PrintService,%20javax.print.attribute.PrintRequestAttributeSet,%20boolean)), [read](http://docs.google.com/javax/swing/text/JTextComponent.html#read(java.io.Reader,%20java.lang.Object)), [removeCaretListener](http://docs.google.com/javax/swing/text/JTextComponent.html#removeCaretListener(javax.swing.event.CaretListener)), [removeKeymap](http://docs.google.com/javax/swing/text/JTextComponent.html#removeKeymap(java.lang.String)), [removeNotify](http://docs.google.com/javax/swing/text/JTextComponent.html#removeNotify()), [replaceSelection](http://docs.google.com/javax/swing/text/JTextComponent.html#replaceSelection(java.lang.String)), [select](http://docs.google.com/javax/swing/text/JTextComponent.html#select(int,%20int)), [selectAll](http://docs.google.com/javax/swing/text/JTextComponent.html#selectAll()), [setCaret](http://docs.google.com/javax/swing/text/JTextComponent.html#setCaret(javax.swing.text.Caret)), [setCaretColor](http://docs.google.com/javax/swing/text/JTextComponent.html#setCaretColor(java.awt.Color)), [setCaretPosition](http://docs.google.com/javax/swing/text/JTextComponent.html#setCaretPosition(int)), [setComponentOrientation](http://docs.google.com/javax/swing/text/JTextComponent.html#setComponentOrientation(java.awt.ComponentOrientation)), [setDisabledTextColor](http://docs.google.com/javax/swing/text/JTextComponent.html#setDisabledTextColor(java.awt.Color)), [setDragEnabled](http://docs.google.com/javax/swing/text/JTextComponent.html#setDragEnabled(boolean)), [setDropMode](http://docs.google.com/javax/swing/text/JTextComponent.html#setDropMode(javax.swing.DropMode)), [setEditable](http://docs.google.com/javax/swing/text/JTextComponent.html#setEditable(boolean)), [setFocusAccelerator](http://docs.google.com/javax/swing/text/JTextComponent.html#setFocusAccelerator(char)), [setHighlighter](http://docs.google.com/javax/swing/text/JTextComponent.html#setHighlighter(javax.swing.text.Highlighter)), [setKeymap](http://docs.google.com/javax/swing/text/JTextComponent.html#setKeymap(javax.swing.text.Keymap)), [setMargin](http://docs.google.com/javax/swing/text/JTextComponent.html#setMargin(java.awt.Insets)), [setNavigationFilter](http://docs.google.com/javax/swing/text/JTextComponent.html#setNavigationFilter(javax.swing.text.NavigationFilter)), [setSelectedTextColor](http://docs.google.com/javax/swing/text/JTextComponent.html#setSelectedTextColor(java.awt.Color)), [setSelectionColor](http://docs.google.com/javax/swing/text/JTextComponent.html#setSelectionColor(java.awt.Color)), [setSelectionEnd](http://docs.google.com/javax/swing/text/JTextComponent.html#setSelectionEnd(int)), [setSelectionStart](http://docs.google.com/javax/swing/text/JTextComponent.html#setSelectionStart(int)), [setText](http://docs.google.com/javax/swing/text/JTextComponent.html#setText(java.lang.String)), [setUI](http://docs.google.com/javax/swing/text/JTextComponent.html#setUI(javax.swing.plaf.TextUI)), [updateUI](http://docs.google.com/javax/swing/text/JTextComponent.html#updateUI()), [viewToModel](http://docs.google.com/javax/swing/text/JTextComponent.html#viewToModel(java.awt.Point)), [write](http://docs.google.com/javax/swing/text/JTextComponent.html#write(java.io.Writer)) |

| **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)), [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()), [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()), [paint](http://docs.google.com/javax/swing/JComponent.html#paint(java.awt.Graphics)), [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)), [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()), [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)), [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()), [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)), [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()), [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)), [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()), [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)), [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)), [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)), [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** |
| --- |

### COMMIT

public static final int **COMMIT**

Constant identifying that when focus is lost, commitEdit should be invoked. If in commiting the new value a ParseException is thrown, the invalid value will remain.

**See Also:**[setFocusLostBehavior(int)](http://docs.google.com/javax/swing/JFormattedTextField.html#setFocusLostBehavior(int)), [Constant Field Values](http://docs.google.com/constant-values.html#javax.swing.JFormattedTextField.COMMIT)

### COMMIT\_OR\_REVERT

public static final int **COMMIT\_OR\_REVERT**

Constant identifying that when focus is lost, commitEdit should be invoked. If in commiting the new value a ParseException is thrown, the value will be reverted.

**See Also:**[setFocusLostBehavior(int)](http://docs.google.com/javax/swing/JFormattedTextField.html#setFocusLostBehavior(int)), [Constant Field Values](http://docs.google.com/constant-values.html#javax.swing.JFormattedTextField.COMMIT_OR_REVERT)

### REVERT

public static final int **REVERT**

Constant identifying that when focus is lost, editing value should be reverted to current value set on the JFormattedTextField.

**See Also:**[setFocusLostBehavior(int)](http://docs.google.com/javax/swing/JFormattedTextField.html#setFocusLostBehavior(int)), [Constant Field Values](http://docs.google.com/constant-values.html#javax.swing.JFormattedTextField.REVERT)

### PERSIST

public static final int **PERSIST**

Constant identifying that when focus is lost, the edited value should be left.

**See Also:**[setFocusLostBehavior(int)](http://docs.google.com/javax/swing/JFormattedTextField.html#setFocusLostBehavior(int)), [Constant Field Values](http://docs.google.com/constant-values.html#javax.swing.JFormattedTextField.PERSIST)

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

### JFormattedTextField

public **JFormattedTextField**()

Creates a JFormattedTextField with no AbstractFormatterFactory. Use setMask or setFormatterFactory to configure the JFormattedTextField to edit a particular type of value.

### JFormattedTextField

public **JFormattedTextField**([Object](http://docs.google.com/java/lang/Object.html) value)

Creates a JFormattedTextField with the specified value. This will create an AbstractFormatterFactory based on the type of value.

**Parameters:**value - Initial value for the JFormattedTextField

### JFormattedTextField

public **JFormattedTextField**([Format](http://docs.google.com/java/text/Format.html) format)

Creates a JFormattedTextField. format is wrapped in an appropriate AbstractFormatter which is then wrapped in an AbstractFormatterFactory.

**Parameters:**format - Format used to look up an AbstractFormatter

### JFormattedTextField

public **JFormattedTextField**([JFormattedTextField.AbstractFormatter](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html) formatter)

Creates a JFormattedTextField with the specified AbstractFormatter. The AbstractFormatter is placed in an AbstractFormatterFactory.

**Parameters:**formatter - AbstractFormatter to use for formatting.

### JFormattedTextField

public **JFormattedTextField**([JFormattedTextField.AbstractFormatterFactory](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatterFactory.html) factory)

Creates a JFormattedTextField with the specified AbstractFormatterFactory.

**Parameters:**factory - AbstractFormatterFactory used for formatting.

### JFormattedTextField

public **JFormattedTextField**([JFormattedTextField.AbstractFormatterFactory](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatterFactory.html) factory,  
 [Object](http://docs.google.com/java/lang/Object.html) currentValue)

Creates a JFormattedTextField with the specified AbstractFormatterFactory and initial value.

**Parameters:**factory - AbstractFormatterFactory used for formatting.currentValue - Initial value to use

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

### setFocusLostBehavior

public void **setFocusLostBehavior**(int behavior)

Sets the behavior when focus is lost. This will be one of JFormattedTextField.COMMIT\_OR\_REVERT, JFormattedTextField.REVERT, JFormattedTextField.COMMIT or JFormattedTextField.PERSIST Note that some AbstractFormatters may push changes as they occur, so that the value of this will have no effect.

This will throw an IllegalArgumentException if the object passed in is not one of the afore mentioned values.

The default value of this property is JFormattedTextField.COMMIT\_OR\_REVERT.

**Parameters:**behavior - Identifies behavior when focus is lost **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if behavior is not one of the known values

### getFocusLostBehavior

public int **getFocusLostBehavior**()

Returns the behavior when focus is lost. This will be one of COMMIT\_OR\_REVERT, COMMIT, REVERT or PERSIST Note that some AbstractFormatters may push changes as they occur, so that the value of this will have no effect.

**Returns:**returns behavior when focus is lost

### setFormatterFactory

public void **setFormatterFactory**([JFormattedTextField.AbstractFormatterFactory](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatterFactory.html) tf)

Sets the AbstractFormatterFactory. AbstractFormatterFactory is able to return an instance of AbstractFormatter that is used to format a value for display, as well an enforcing an editing policy.

If you have not explicitly set an AbstractFormatterFactory by way of this method (or a constructor) an AbstractFormatterFactory and consequently an AbstractFormatter will be used based on the Class of the value. NumberFormatter will be used for Numbers, DateFormatter will be used for Dates, otherwise DefaultFormatter will be used.

This is a JavaBeans bound property.

**Parameters:**tf - AbstractFormatterFactory used to lookup instances of AbstractFormatter

### getFormatterFactory

public [JFormattedTextField.AbstractFormatterFactory](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatterFactory.html) **getFormatterFactory**()

Returns the current AbstractFormatterFactory.

**Returns:**AbstractFormatterFactory used to determine AbstractFormatters**See Also:**[setFormatterFactory(javax.swing.JFormattedTextField.AbstractFormatterFactory)](http://docs.google.com/javax/swing/JFormattedTextField.html#setFormatterFactory(javax.swing.JFormattedTextField.AbstractFormatterFactory))

### setFormatter

protected void **setFormatter**([JFormattedTextField.AbstractFormatter](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html) format)

Sets the current AbstractFormatter.

You should not normally invoke this, instead set the AbstractFormatterFactory or set the value. JFormattedTextField will invoke this as the state of the JFormattedTextField changes and requires the value to be reset. JFormattedTextField passes in the AbstractFormatter obtained from the AbstractFormatterFactory.

This is a JavaBeans bound property.

**Parameters:**format - AbstractFormatter to use for formatting**See Also:**[setFormatterFactory(javax.swing.JFormattedTextField.AbstractFormatterFactory)](http://docs.google.com/javax/swing/JFormattedTextField.html#setFormatterFactory(javax.swing.JFormattedTextField.AbstractFormatterFactory))

### getFormatter

public [JFormattedTextField.AbstractFormatter](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html) **getFormatter**()

Returns the AbstractFormatter that is used to format and parse the current value.

**Returns:**AbstractFormatter used for formatting

### setValue

public void **setValue**([Object](http://docs.google.com/java/lang/Object.html) value)

Sets the value that will be formatted by an AbstractFormatter obtained from the current AbstractFormatterFactory. If no AbstractFormatterFactory has been specified, this will attempt to create one based on the type of value.

The default value of this property is null.

This is a JavaBeans bound property.

**Parameters:**value - Current value to display

### getValue

public [Object](http://docs.google.com/java/lang/Object.html) **getValue**()

Returns the last valid value. Based on the editing policy of the AbstractFormatter this may not return the current value. The currently edited value can be obtained by invoking commitEdit followed by getValue.

**Returns:**Last valid value

### commitEdit

public void **commitEdit**()  
 throws [ParseException](http://docs.google.com/java/text/ParseException.html)

Forces the current value to be taken from the AbstractFormatter and set as the current value. This has no effect if there is no current AbstractFormatter installed.

**Throws:** [ParseException](http://docs.google.com/java/text/ParseException.html) - if the AbstractFormatter is not able to format the current value

### isEditValid

public boolean **isEditValid**()

Returns true if the current value being edited is valid. The value of this is managed by the current AbstractFormatter, as such there is no public setter for it.

**Returns:**true if the current value being edited is valid.

### invalidEdit

protected void **invalidEdit**()

Invoked when the user inputs an invalid value. This gives the component a chance to provide feedback. The default implementation beeps.

### processInputMethodEvent

protected void **processInputMethodEvent**([InputMethodEvent](http://docs.google.com/java/awt/event/InputMethodEvent.html) e)

Processes any input method events, such as InputMethodEvent.INPUT\_METHOD\_TEXT\_CHANGED or InputMethodEvent.CARET\_POSITION\_CHANGED.

**Overrides:**[processInputMethodEvent](http://docs.google.com/javax/swing/text/JTextComponent.html#processInputMethodEvent(java.awt.event.InputMethodEvent)) in class [JTextComponent](http://docs.google.com/javax/swing/text/JTextComponent.html) **Parameters:**e - the InputMethodEvent**See Also:**[InputMethodEvent](http://docs.google.com/java/awt/event/InputMethodEvent.html)

### processFocusEvent

protected void **processFocusEvent**([FocusEvent](http://docs.google.com/java/awt/event/FocusEvent.html) e)

Processes any focus events, such as FocusEvent.FOCUS\_GAINED or FocusEvent.FOCUS\_LOST.

**Overrides:**[processFocusEvent](http://docs.google.com/java/awt/Component.html#processFocusEvent(java.awt.event.FocusEvent)) in class [Component](http://docs.google.com/java/awt/Component.html) **Parameters:**e - the FocusEvent**See Also:**[FocusEvent](http://docs.google.com/java/awt/event/FocusEvent.html)

### getActions

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

Fetches the command list for the editor. This is the list of commands supported by the plugged-in UI augmented by the collection of commands that the editor itself supports. These are useful for binding to events, such as in a keymap.

**Overrides:**[getActions](http://docs.google.com/javax/swing/JTextField.html#getActions()) in class [JTextField](http://docs.google.com/javax/swing/JTextField.html) **Returns:**the command list

### getUIClassID

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

Gets the class ID for a UI.

**Overrides:**[getUIClassID](http://docs.google.com/javax/swing/JTextField.html#getUIClassID()) in class [JTextField](http://docs.google.com/javax/swing/JTextField.html) **Returns:**the string "FormattedTextFieldUI"**See Also:**[JComponent.getUIClassID()](http://docs.google.com/javax/swing/JComponent.html#getUIClassID())

### setDocument

public void **setDocument**([Document](http://docs.google.com/javax/swing/text/Document.html) doc)

Associates the editor with a text document. The currently registered factory is used to build a view for the document, which gets displayed by the editor after revalidation. A PropertyChange event ("document") is propagated to each listener.

**Overrides:**[setDocument](http://docs.google.com/javax/swing/JTextField.html#setDocument(javax.swing.text.Document)) in class [JTextField](http://docs.google.com/javax/swing/JTextField.html) **Parameters:**doc - the document to display/edit**See Also:**[JTextComponent.getDocument()](http://docs.google.com/javax/swing/text/JTextComponent.html#getDocument())

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/JFormattedTextField.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/JFileChooser.AccessibleJFileChooser.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/JFormattedTextField.html)    [**NO FRAMES**](http://docs.google.com/JFormattedTextField.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | [FIELD](#2s8eyo1) | [CONSTR](#44sinio) | [METHOD](#2jxsxqh) | DETAIL: [FIELD](#3whwml4) | [CONSTR](#49x2ik5) | [METHOD](#1hmsyys) |

[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).