| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/NumberFormatter.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/text/NavigationFilter.FilterBypass.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/ParagraphView.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/NumberFormatter.html)    [**NO FRAMES**](http://docs.google.com/NumberFormatter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#26in1rg) |

## **javax.swing.text**

Class NumberFormatter

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [javax.swing.JFormattedTextField.AbstractFormatter](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html)  
 [javax.swing.text.DefaultFormatter](http://docs.google.com/javax/swing/text/DefaultFormatter.html)  
 [javax.swing.text.InternationalFormatter](http://docs.google.com/javax/swing/text/InternationalFormatter.html)  
 **javax.swing.text.NumberFormatter**

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [Cloneable](http://docs.google.com/java/lang/Cloneable.html)

public class **NumberFormatter**extends [InternationalFormatter](http://docs.google.com/javax/swing/text/InternationalFormatter.html)

NumberFormatter subclasses InternationalFormatter adding special behavior for numbers. Among the specializations are (these are only used if the NumberFormatter does not display invalid nubers, eg setAllowsInvalid(false)):

* Pressing +/- (- is determined from the DecimalFormatSymbols associated with the DecimalFormat) in any field but the exponent field will attempt to change the sign of the number to positive/negative.
* Pressing +/- (- is determined from the DecimalFormatSymbols associated with the DecimalFormat) in the exponent field will attemp to change the sign of the exponent to positive/negative.

If you are displaying scientific numbers, you may wish to turn on overwrite mode, setOverwriteMode(true). For example:

DecimalFormat decimalFormat = new DecimalFormat("0.000E0");  
 NumberFormatter textFormatter = new NumberFormatter(decimalFormat);  
 textFormatter.setOverwriteMode(true);  
 textFormatter.setAllowsInvalid(false);

If you are going to allow the user to enter decimal values, you should either force the DecimalFormat to contain at least one decimal (#.0###), or allow the value to be invalid setAllowsInvalid(true). Otherwise users may not be able to input decimal values.

NumberFormatter provides slightly different behavior to stringToValue than that of its superclass. If you have specified a Class for values, [DefaultFormatter.setValueClass(java.lang.Class)](http://docs.google.com/javax/swing/text/DefaultFormatter.html#setValueClass(java.lang.Class)), that is one of of Integer, Long, Float, Double, Byte or Short and the Format's parseObject returns an instance of Number, the corresponding instance of the value class will be created using the constructor appropriate for the primitive type the value class represents. For example: setValueClass(Integer.class) will cause the resulting value to be created via new Integer(((Number)formatter.parseObject(string)).intValue()). This is typically useful if you wish to set a min/max value as the various Number implementations are generally not comparable to each other. This is also useful if for some reason you need a specific Number implementation for your values.

**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

| **Constructor Summary** | |
| --- | --- |
| [**NumberFormatter**](http://docs.google.com/javax/swing/text/NumberFormatter.html#NumberFormatter())()            Creates a NumberFormatter with the a default NumberFormat instance obtained from NumberFormat.getNumberInstance(). |
| [**NumberFormatter**](http://docs.google.com/javax/swing/text/NumberFormatter.html#NumberFormatter(java.text.NumberFormat))([NumberFormat](http://docs.google.com/java/text/NumberFormat.html) format)            Creates a NumberFormatter with the specified Format instance. |

| **Method Summary** | |
| --- | --- |
| void | [**setFormat**](http://docs.google.com/javax/swing/text/NumberFormatter.html#setFormat(java.text.Format))([Format](http://docs.google.com/java/text/Format.html) format)            Sets the format that dictates the legal values that can be edited and displayed. |

| **Methods inherited from class javax.swing.text.**[**InternationalFormatter**](http://docs.google.com/javax/swing/text/InternationalFormatter.html) |
| --- |
| [clone](http://docs.google.com/javax/swing/text/InternationalFormatter.html#clone()), [getActions](http://docs.google.com/javax/swing/text/InternationalFormatter.html#getActions()), [getFields](http://docs.google.com/javax/swing/text/InternationalFormatter.html#getFields(int)), [getFormat](http://docs.google.com/javax/swing/text/InternationalFormatter.html#getFormat()), [getMaximum](http://docs.google.com/javax/swing/text/InternationalFormatter.html#getMaximum()), [getMinimum](http://docs.google.com/javax/swing/text/InternationalFormatter.html#getMinimum()), [install](http://docs.google.com/javax/swing/text/InternationalFormatter.html#install(javax.swing.JFormattedTextField)), [setMaximum](http://docs.google.com/javax/swing/text/InternationalFormatter.html#setMaximum(java.lang.Comparable)), [setMinimum](http://docs.google.com/javax/swing/text/InternationalFormatter.html#setMinimum(java.lang.Comparable)), [stringToValue](http://docs.google.com/javax/swing/text/InternationalFormatter.html#stringToValue(java.lang.String)), [valueToString](http://docs.google.com/javax/swing/text/InternationalFormatter.html#valueToString(java.lang.Object)) |

| **Methods inherited from class javax.swing.text.**[**DefaultFormatter**](http://docs.google.com/javax/swing/text/DefaultFormatter.html) |
| --- |
| [getAllowsInvalid](http://docs.google.com/javax/swing/text/DefaultFormatter.html#getAllowsInvalid()), [getCommitsOnValidEdit](http://docs.google.com/javax/swing/text/DefaultFormatter.html#getCommitsOnValidEdit()), [getDocumentFilter](http://docs.google.com/javax/swing/text/DefaultFormatter.html#getDocumentFilter()), [getNavigationFilter](http://docs.google.com/javax/swing/text/DefaultFormatter.html#getNavigationFilter()), [getOverwriteMode](http://docs.google.com/javax/swing/text/DefaultFormatter.html#getOverwriteMode()), [getValueClass](http://docs.google.com/javax/swing/text/DefaultFormatter.html#getValueClass()), [setAllowsInvalid](http://docs.google.com/javax/swing/text/DefaultFormatter.html#setAllowsInvalid(boolean)), [setCommitsOnValidEdit](http://docs.google.com/javax/swing/text/DefaultFormatter.html#setCommitsOnValidEdit(boolean)), [setOverwriteMode](http://docs.google.com/javax/swing/text/DefaultFormatter.html#setOverwriteMode(boolean)), [setValueClass](http://docs.google.com/javax/swing/text/DefaultFormatter.html#setValueClass(java.lang.Class)) |

| **Methods inherited from class javax.swing.**[**JFormattedTextField.AbstractFormatter**](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html) |
| --- |
| [getFormattedTextField](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html#getFormattedTextField()), [invalidEdit](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html#invalidEdit()), [setEditValid](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html#setEditValid(boolean)), [uninstall](http://docs.google.com/javax/swing/JFormattedTextField.AbstractFormatter.html#uninstall()) |

| **Methods inherited from class java.lang.**[**Object**](http://docs.google.com/java/lang/Object.html) |
| --- |
| [equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [hashCode](http://docs.google.com/java/lang/Object.html#hashCode()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [toString](http://docs.google.com/java/lang/Object.html#toString()), [wait](http://docs.google.com/java/lang/Object.html#wait()), [wait](http://docs.google.com/java/lang/Object.html#wait(long)), [wait](http://docs.google.com/java/lang/Object.html#wait(long,%20int)) |

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

### NumberFormatter

public **NumberFormatter**()

Creates a NumberFormatter with the a default NumberFormat instance obtained from NumberFormat.getNumberInstance().

### NumberFormatter

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

Creates a NumberFormatter with the specified Format instance.

**Parameters:**format - Format used to dictate legal values

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

### setFormat

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

Sets the format that dictates the legal values that can be edited and displayed.

If you have used the nullary constructor the value of this property will be determined for the current locale by way of the NumberFormat.getNumberInstance() method.

**Overrides:**[setFormat](http://docs.google.com/javax/swing/text/InternationalFormatter.html#setFormat(java.text.Format)) in class [InternationalFormatter](http://docs.google.com/javax/swing/text/InternationalFormatter.html) **Parameters:**format - NumberFormat instance used to dictate legal values

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/NumberFormatter.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/text/NavigationFilter.FilterBypass.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/ParagraphView.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/NumberFormatter.html)    [**NO FRAMES**](http://docs.google.com/NumberFormatter.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#26in1rg) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).