| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DocumentFilter.FilterBypass.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/DocumentFilter.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/EditorKit.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/DocumentFilter.FilterBypass.html)    [**NO FRAMES**](http://docs.google.com/DocumentFilter.FilterBypass.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

## **javax.swing.text**

Class DocumentFilter.FilterBypass

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **javax.swing.text.DocumentFilter.FilterBypass**

**Enclosing class:**[DocumentFilter](http://docs.google.com/javax/swing/text/DocumentFilter.html)

public abstract static class **DocumentFilter.FilterBypass**extends [Object](http://docs.google.com/java/lang/Object.html)

Used as a way to circumvent calling back into the Document to change it. Document implementations that wish to support a DocumentFilter must provide an implementation that will not callback into the DocumentFilter when the following methods are invoked from the DocumentFilter.

**Since:** 1.4

| **Constructor Summary** | |
| --- | --- |
| [**DocumentFilter.FilterBypass**](http://docs.google.com/javax/swing/text/DocumentFilter.FilterBypass.html#DocumentFilter.FilterBypass())() |

| **Method Summary** | |
| --- | --- |
| abstract  [Document](http://docs.google.com/javax/swing/text/Document.html) | [**getDocument**](http://docs.google.com/javax/swing/text/DocumentFilter.FilterBypass.html#getDocument())()            Returns the Document the mutation is occuring on. |
| abstract  void | [**insertString**](http://docs.google.com/javax/swing/text/DocumentFilter.FilterBypass.html#insertString(int,%20java.lang.String,%20javax.swing.text.AttributeSet))(int offset, [String](http://docs.google.com/java/lang/String.html) string, [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attr)            Inserts the specified text, bypassing the DocumentFilter. |
| abstract  void | [**remove**](http://docs.google.com/javax/swing/text/DocumentFilter.FilterBypass.html#remove(int,%20int))(int offset, int length)            Removes the specified region of text, bypassing the DocumentFilter. |
| abstract  void | [**replace**](http://docs.google.com/javax/swing/text/DocumentFilter.FilterBypass.html#replace(int,%20int,%20java.lang.String,%20javax.swing.text.AttributeSet))(int offset, int length, [String](http://docs.google.com/java/lang/String.html) string, [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attrs)            Deletes the region of text from offset to offset + length, and replaces it with text. |

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

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

### DocumentFilter.FilterBypass

public **DocumentFilter.FilterBypass**()

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

### getDocument

public abstract [Document](http://docs.google.com/javax/swing/text/Document.html) **getDocument**()

Returns the Document the mutation is occuring on.

**Returns:**Document that remove/insertString will operate on

### remove

public abstract void **remove**(int offset,  
 int length)  
 throws [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html)

Removes the specified region of text, bypassing the DocumentFilter.

**Parameters:**offset - the offset from the beginning >= 0length - the number of characters to remove >= 0 **Throws:** [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html) - some portion of the removal range was not a valid part of the document. The location in the exception is the first bad position encountered.

### insertString

public abstract void **insertString**(int offset,  
 [String](http://docs.google.com/java/lang/String.html) string,  
 [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attr)  
 throws [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html)

Inserts the specified text, bypassing the DocumentFilter.

**Parameters:**offset - the offset into the document to insert the content >= 0. All positions that track change at or after the given location will move.string - the string to insertattr - the attributes to associate with the inserted content. This may be null if there are no attributes. **Throws:** [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html) - the given insert position is not a valid position within the document

### replace

public abstract void **replace**(int offset,  
 int length,  
 [String](http://docs.google.com/java/lang/String.html) string,  
 [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attrs)  
 throws [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html)

Deletes the region of text from offset to offset + length, and replaces it with text.

**Parameters:**offset - Location in Documentlength - Length of text to deletestring - Text to insert, null indicates no text to insertattrs - AttributeSet indicating attributes of inserted text, null is legal. **Throws:** [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html) - the given insert is not a valid position within the document

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DocumentFilter.FilterBypass.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/DocumentFilter.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/EditorKit.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/DocumentFilter.FilterBypass.html)    [**NO FRAMES**](http://docs.google.com/DocumentFilter.FilterBypass.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#4d34og8) |

[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).