| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/PlainDocument.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/PasswordView.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/PlainView.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/PlainDocument.html)    [**NO FRAMES**](http://docs.google.com/PlainDocument.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#2et92p0) | [FIELD](#tyjcwt) | [CONSTR](#4d34og8) | [METHOD](#2s8eyo1) | DETAIL: [FIELD](#26in1rg) | [CONSTR](#1ksv4uv) | [METHOD](#z337ya) |

## **javax.swing.text**

Class PlainDocument

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

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [Document](http://docs.google.com/javax/swing/text/Document.html)

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

A plain document that maintains no character attributes. The default element structure for this document is a map of the lines in the text. The Element returned by getDefaultRootElement is a map of the lines, and each child element represents a line. This model does not maintain any character level attributes, but each line can be tagged with an arbitrary set of attributes. Line to offset, and offset to line translations can be quickly performed using the default root element. The structure information of the DocumentEvent's fired by edits will indicate the line structure changes.

The default content storage management is performed by a gapped buffer implementation (GapContent). It supports editing reasonably large documents with good efficiency when the edits are contiguous or clustered, as is typical.

**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).

**See Also:**[Document](http://docs.google.com/javax/swing/text/Document.html), [AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html)

| **Nested Class Summary** | |
| --- | --- |

| **Nested classes/interfaces inherited from class javax.swing.text.**[**AbstractDocument**](http://docs.google.com/javax/swing/text/AbstractDocument.html) |
| --- |
| [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html), [AbstractDocument.AttributeContext](http://docs.google.com/javax/swing/text/AbstractDocument.AttributeContext.html), [AbstractDocument.BranchElement](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html), [AbstractDocument.Content](http://docs.google.com/javax/swing/text/AbstractDocument.Content.html), [AbstractDocument.DefaultDocumentEvent](http://docs.google.com/javax/swing/text/AbstractDocument.DefaultDocumentEvent.html), [AbstractDocument.ElementEdit](http://docs.google.com/javax/swing/text/AbstractDocument.ElementEdit.html), [AbstractDocument.LeafElement](http://docs.google.com/javax/swing/text/AbstractDocument.LeafElement.html) |

| **Field Summary** | |
| --- | --- |
| static [String](http://docs.google.com/java/lang/String.html) | [**lineLimitAttribute**](http://docs.google.com/javax/swing/text/PlainDocument.html#lineLimitAttribute)            Name of the attribute that specifies the maximum length of a line, if there is a maximum length. |
| static [String](http://docs.google.com/java/lang/String.html) | [**tabSizeAttribute**](http://docs.google.com/javax/swing/text/PlainDocument.html#tabSizeAttribute)            Name of the attribute that specifies the tab size for tabs contained in the content. |

| **Fields inherited from class javax.swing.text.**[**AbstractDocument**](http://docs.google.com/javax/swing/text/AbstractDocument.html) |
| --- |
| [BAD\_LOCATION](http://docs.google.com/javax/swing/text/AbstractDocument.html#BAD_LOCATION), [BidiElementName](http://docs.google.com/javax/swing/text/AbstractDocument.html#BidiElementName), [ContentElementName](http://docs.google.com/javax/swing/text/AbstractDocument.html#ContentElementName), [ElementNameAttribute](http://docs.google.com/javax/swing/text/AbstractDocument.html#ElementNameAttribute), [listenerList](http://docs.google.com/javax/swing/text/AbstractDocument.html#listenerList), [ParagraphElementName](http://docs.google.com/javax/swing/text/AbstractDocument.html#ParagraphElementName), [SectionElementName](http://docs.google.com/javax/swing/text/AbstractDocument.html#SectionElementName) |

| **Fields inherited from interface javax.swing.text.**[**Document**](http://docs.google.com/javax/swing/text/Document.html) |
| --- |
| [StreamDescriptionProperty](http://docs.google.com/javax/swing/text/Document.html#StreamDescriptionProperty), [TitleProperty](http://docs.google.com/javax/swing/text/Document.html#TitleProperty) |

| **Constructor Summary** | |
| --- | --- |
| [**PlainDocument**](http://docs.google.com/javax/swing/text/PlainDocument.html#PlainDocument())()            Constructs a plain text document. |
| [**PlainDocument**](http://docs.google.com/javax/swing/text/PlainDocument.html#PlainDocument(javax.swing.text.AbstractDocument.Content))([AbstractDocument.Content](http://docs.google.com/javax/swing/text/AbstractDocument.Content.html) c)            Constructs a plain text document. |

| **Method Summary** | |
| --- | --- |
| protected  [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) | [**createDefaultRoot**](http://docs.google.com/javax/swing/text/PlainDocument.html#createDefaultRoot())()            Creates the root element to be used to represent the default document structure. |
| [Element](http://docs.google.com/javax/swing/text/Element.html) | [**getDefaultRootElement**](http://docs.google.com/javax/swing/text/PlainDocument.html#getDefaultRootElement())()            Gets the default root element for the document model. |
| [Element](http://docs.google.com/javax/swing/text/Element.html) | [**getParagraphElement**](http://docs.google.com/javax/swing/text/PlainDocument.html#getParagraphElement(int))(int pos)            Get the paragraph element containing the given position. |
| void | [**insertString**](http://docs.google.com/javax/swing/text/PlainDocument.html#insertString(int,%20java.lang.String,%20javax.swing.text.AttributeSet))(int offs, [String](http://docs.google.com/java/lang/String.html) str, [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) a)            Inserts some content into the document. |
| protected  void | [**insertUpdate**](http://docs.google.com/javax/swing/text/PlainDocument.html#insertUpdate(javax.swing.text.AbstractDocument.DefaultDocumentEvent,%20javax.swing.text.AttributeSet))([AbstractDocument.DefaultDocumentEvent](http://docs.google.com/javax/swing/text/AbstractDocument.DefaultDocumentEvent.html) chng, [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attr)            Updates document structure as a result of text insertion. |
| protected  void | [**removeUpdate**](http://docs.google.com/javax/swing/text/PlainDocument.html#removeUpdate(javax.swing.text.AbstractDocument.DefaultDocumentEvent))([AbstractDocument.DefaultDocumentEvent](http://docs.google.com/javax/swing/text/AbstractDocument.DefaultDocumentEvent.html) chng)            Updates any document structure as a result of text removal. |

| **Methods inherited from class javax.swing.text.**[**AbstractDocument**](http://docs.google.com/javax/swing/text/AbstractDocument.html) |
| --- |
| [addDocumentListener](http://docs.google.com/javax/swing/text/AbstractDocument.html#addDocumentListener(javax.swing.event.DocumentListener)), [addUndoableEditListener](http://docs.google.com/javax/swing/text/AbstractDocument.html#addUndoableEditListener(javax.swing.event.UndoableEditListener)), [createBranchElement](http://docs.google.com/javax/swing/text/AbstractDocument.html#createBranchElement(javax.swing.text.Element,%20javax.swing.text.AttributeSet)), [createLeafElement](http://docs.google.com/javax/swing/text/AbstractDocument.html#createLeafElement(javax.swing.text.Element,%20javax.swing.text.AttributeSet,%20int,%20int)), [createPosition](http://docs.google.com/javax/swing/text/AbstractDocument.html#createPosition(int)), [dump](http://docs.google.com/javax/swing/text/AbstractDocument.html#dump(java.io.PrintStream)), [fireChangedUpdate](http://docs.google.com/javax/swing/text/AbstractDocument.html#fireChangedUpdate(javax.swing.event.DocumentEvent)), [fireInsertUpdate](http://docs.google.com/javax/swing/text/AbstractDocument.html#fireInsertUpdate(javax.swing.event.DocumentEvent)), [fireRemoveUpdate](http://docs.google.com/javax/swing/text/AbstractDocument.html#fireRemoveUpdate(javax.swing.event.DocumentEvent)), [fireUndoableEditUpdate](http://docs.google.com/javax/swing/text/AbstractDocument.html#fireUndoableEditUpdate(javax.swing.event.UndoableEditEvent)), [getAsynchronousLoadPriority](http://docs.google.com/javax/swing/text/AbstractDocument.html#getAsynchronousLoadPriority()), [getAttributeContext](http://docs.google.com/javax/swing/text/AbstractDocument.html#getAttributeContext()), [getBidiRootElement](http://docs.google.com/javax/swing/text/AbstractDocument.html#getBidiRootElement()), [getContent](http://docs.google.com/javax/swing/text/AbstractDocument.html#getContent()), [getCurrentWriter](http://docs.google.com/javax/swing/text/AbstractDocument.html#getCurrentWriter()), [getDocumentFilter](http://docs.google.com/javax/swing/text/AbstractDocument.html#getDocumentFilter()), [getDocumentListeners](http://docs.google.com/javax/swing/text/AbstractDocument.html#getDocumentListeners()), [getDocumentProperties](http://docs.google.com/javax/swing/text/AbstractDocument.html#getDocumentProperties()), [getEndPosition](http://docs.google.com/javax/swing/text/AbstractDocument.html#getEndPosition()), [getLength](http://docs.google.com/javax/swing/text/AbstractDocument.html#getLength()), [getListeners](http://docs.google.com/javax/swing/text/AbstractDocument.html#getListeners(java.lang.Class)), [getProperty](http://docs.google.com/javax/swing/text/AbstractDocument.html#getProperty(java.lang.Object)), [getRootElements](http://docs.google.com/javax/swing/text/AbstractDocument.html#getRootElements()), [getStartPosition](http://docs.google.com/javax/swing/text/AbstractDocument.html#getStartPosition()), [getText](http://docs.google.com/javax/swing/text/AbstractDocument.html#getText(int,%20int)), [getText](http://docs.google.com/javax/swing/text/AbstractDocument.html#getText(int,%20int,%20javax.swing.text.Segment)), [getUndoableEditListeners](http://docs.google.com/javax/swing/text/AbstractDocument.html#getUndoableEditListeners()), [postRemoveUpdate](http://docs.google.com/javax/swing/text/AbstractDocument.html#postRemoveUpdate(javax.swing.text.AbstractDocument.DefaultDocumentEvent)), [putProperty](http://docs.google.com/javax/swing/text/AbstractDocument.html#putProperty(java.lang.Object,%20java.lang.Object)), [readLock](http://docs.google.com/javax/swing/text/AbstractDocument.html#readLock()), [readUnlock](http://docs.google.com/javax/swing/text/AbstractDocument.html#readUnlock()), [remove](http://docs.google.com/javax/swing/text/AbstractDocument.html#remove(int,%20int)), [removeDocumentListener](http://docs.google.com/javax/swing/text/AbstractDocument.html#removeDocumentListener(javax.swing.event.DocumentListener)), [removeUndoableEditListener](http://docs.google.com/javax/swing/text/AbstractDocument.html#removeUndoableEditListener(javax.swing.event.UndoableEditListener)), [render](http://docs.google.com/javax/swing/text/AbstractDocument.html#render(java.lang.Runnable)), [replace](http://docs.google.com/javax/swing/text/AbstractDocument.html#replace(int,%20int,%20java.lang.String,%20javax.swing.text.AttributeSet)), [setAsynchronousLoadPriority](http://docs.google.com/javax/swing/text/AbstractDocument.html#setAsynchronousLoadPriority(int)), [setDocumentFilter](http://docs.google.com/javax/swing/text/AbstractDocument.html#setDocumentFilter(javax.swing.text.DocumentFilter)), [setDocumentProperties](http://docs.google.com/javax/swing/text/AbstractDocument.html#setDocumentProperties(java.util.Dictionary)), [writeLock](http://docs.google.com/javax/swing/text/AbstractDocument.html#writeLock()), [writeUnlock](http://docs.google.com/javax/swing/text/AbstractDocument.html#writeUnlock()) |

| **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)) |

| **Field Detail** |
| --- |

### tabSizeAttribute

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

Name of the attribute that specifies the tab size for tabs contained in the content. The type for the value is Integer.

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

### lineLimitAttribute

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

Name of the attribute that specifies the maximum length of a line, if there is a maximum length. The type for the value is Integer.

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

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

### PlainDocument

public **PlainDocument**()

Constructs a plain text document. A default model using GapContent is constructed and set.

### PlainDocument

public **PlainDocument**([AbstractDocument.Content](http://docs.google.com/javax/swing/text/AbstractDocument.Content.html) c)

Constructs a plain text document. A default root element is created, and the tab size set to 8.

**Parameters:**c - the container for the content

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

### insertString

public void **insertString**(int offs,  
 [String](http://docs.google.com/java/lang/String.html) str,  
 [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) a)  
 throws [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html)

Inserts some content into the document. Inserting content causes a write lock to be held while the actual changes are taking place, followed by notification to the observers on the thread that grabbed the write lock.

This method is thread safe, although most Swing methods are not. Please see [How to Use Threads](http://java.sun.com/docs/books/tutorial/uiswing/misc/threads.html) for more information.

**Specified by:**[insertString](http://docs.google.com/javax/swing/text/Document.html#insertString(int,%20java.lang.String,%20javax.swing.text.AttributeSet)) in interface [Document](http://docs.google.com/javax/swing/text/Document.html)**Overrides:**[insertString](http://docs.google.com/javax/swing/text/AbstractDocument.html#insertString(int,%20java.lang.String,%20javax.swing.text.AttributeSet)) in class [AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html) **Parameters:**offs - the starting offset >= 0str - the string to insert; does nothing with null/empty stringsa - the attributes for the inserted content **Throws:** [BadLocationException](http://docs.google.com/javax/swing/text/BadLocationException.html) - the given insert position is not a valid position within the document**See Also:**[Document.insertString(int, java.lang.String, javax.swing.text.AttributeSet)](http://docs.google.com/javax/swing/text/Document.html#insertString(int,%20java.lang.String,%20javax.swing.text.AttributeSet))

### getDefaultRootElement

public [Element](http://docs.google.com/javax/swing/text/Element.html) **getDefaultRootElement**()

Gets the default root element for the document model.

**Specified by:**[getDefaultRootElement](http://docs.google.com/javax/swing/text/Document.html#getDefaultRootElement()) in interface [Document](http://docs.google.com/javax/swing/text/Document.html)**Specified by:**[getDefaultRootElement](http://docs.google.com/javax/swing/text/AbstractDocument.html#getDefaultRootElement()) in class [AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html) **Returns:**the root**See Also:**[Document.getDefaultRootElement()](http://docs.google.com/javax/swing/text/Document.html#getDefaultRootElement())

### createDefaultRoot

protected [AbstractDocument.AbstractElement](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html) **createDefaultRoot**()

Creates the root element to be used to represent the default document structure.

**Returns:**the element base

### getParagraphElement

public [Element](http://docs.google.com/javax/swing/text/Element.html) **getParagraphElement**(int pos)

Get the paragraph element containing the given position. Since this document only models lines, it returns the line instead.

**Specified by:**[getParagraphElement](http://docs.google.com/javax/swing/text/AbstractDocument.html#getParagraphElement(int)) in class [AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html) **Parameters:**pos - the starting offset >= 0 **Returns:**the element

### insertUpdate

protected void **insertUpdate**([AbstractDocument.DefaultDocumentEvent](http://docs.google.com/javax/swing/text/AbstractDocument.DefaultDocumentEvent.html) chng,  
 [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attr)

Updates document structure as a result of text insertion. This will happen within a write lock. Since this document simply maps out lines, we refresh the line map.

**Overrides:**[insertUpdate](http://docs.google.com/javax/swing/text/AbstractDocument.html#insertUpdate(javax.swing.text.AbstractDocument.DefaultDocumentEvent,%20javax.swing.text.AttributeSet)) in class [AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html) **Parameters:**chng - the change event describing the ditattr - the set of attributes for the inserted text

### removeUpdate

protected void **removeUpdate**([AbstractDocument.DefaultDocumentEvent](http://docs.google.com/javax/swing/text/AbstractDocument.DefaultDocumentEvent.html) chng)

Updates any document structure as a result of text removal. This will happen within a write lock. Since the structure represents a line map, this just checks to see if the removal spans lines. If it does, the two lines outside of the removal area are joined together.

**Overrides:**[removeUpdate](http://docs.google.com/javax/swing/text/AbstractDocument.html#removeUpdate(javax.swing.text.AbstractDocument.DefaultDocumentEvent)) in class [AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html) **Parameters:**chng - the change event describing the edit

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/PlainDocument.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/PasswordView.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/PlainView.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/PlainDocument.html)    [**NO FRAMES**](http://docs.google.com/PlainDocument.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#2et92p0) | [FIELD](#tyjcwt) | [CONSTR](#4d34og8) | [METHOD](#2s8eyo1) | DETAIL: [FIELD](#26in1rg) | [CONSTR](#1ksv4uv) | [METHOD](#z337ya) |

[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).