| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AbstractDocument.AbstractElement.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/AbstractDocument.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/AbstractDocument.AttributeContext.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/AbstractDocument.AbstractElement.html)    [**NO FRAMES**](http://docs.google.com/AbstractDocument.AbstractElement.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#1t3h5sf) | [METHOD](#4d34og8) | DETAIL: FIELD | [CONSTR](#17dp8vu) | [METHOD](#26in1rg) |

## **javax.swing.text**

Class AbstractDocument.AbstractElement

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

**All Implemented Interfaces:** [Serializable](http://docs.google.com/java/io/Serializable.html), [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html), [Element](http://docs.google.com/javax/swing/text/Element.html), [MutableAttributeSet](http://docs.google.com/javax/swing/text/MutableAttributeSet.html), [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) **Direct Known Subclasses:** [AbstractDocument.BranchElement](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html), [AbstractDocument.LeafElement](http://docs.google.com/javax/swing/text/AbstractDocument.LeafElement.html) **Enclosing class:**[AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html)

public abstract class **AbstractDocument.AbstractElement**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Element](http://docs.google.com/javax/swing/text/Element.html), [MutableAttributeSet](http://docs.google.com/javax/swing/text/MutableAttributeSet.html), [Serializable](http://docs.google.com/java/io/Serializable.html), [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)

Implements the abstract part of an element. By default elements support attributes by having a field that represents the immutable part of the current attribute set for the element. The element itself implements MutableAttributeSet which can be used to modify the set by fetching a new immutable set. The immutable sets are provided by the AttributeContext associated with the document.

**Warning:** Serialized objects of this class will not be compatible with future Swing releases. The current serialization support is appropriate for short term storage or RMI between applications running the same version of Swing. As of 1.4, support for long term storage of all JavaBeansTM has been added to the java.beans package. Please see [XMLEncoder](http://docs.google.com/java/beans/XMLEncoder.html).

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

| **Nested classes/interfaces inherited from interface javax.swing.text.**[**AttributeSet**](http://docs.google.com/javax/swing/text/AttributeSet.html) |
| --- |
| [AttributeSet.CharacterAttribute](http://docs.google.com/javax/swing/text/AttributeSet.CharacterAttribute.html), [AttributeSet.ColorAttribute](http://docs.google.com/javax/swing/text/AttributeSet.ColorAttribute.html), [AttributeSet.FontAttribute](http://docs.google.com/javax/swing/text/AttributeSet.FontAttribute.html), [AttributeSet.ParagraphAttribute](http://docs.google.com/javax/swing/text/AttributeSet.ParagraphAttribute.html) |

| **Field Summary** | |
| --- | --- |

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

| **Constructor Summary** | |
| --- | --- |
| [**AbstractDocument.AbstractElement**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#AbstractDocument.AbstractElement(javax.swing.text.Element,%20javax.swing.text.AttributeSet))([Element](http://docs.google.com/javax/swing/text/Element.html) parent, [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) a)            Creates a new AbstractElement. |

| **Method Summary** | |
| --- | --- |
| void | [**addAttribute**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#addAttribute(java.lang.Object,%20java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) name, [Object](http://docs.google.com/java/lang/Object.html) value)            Adds an attribute to the element. |
| void | [**addAttributes**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#addAttributes(javax.swing.text.AttributeSet))([AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attr)            Adds a set of attributes to the element. |
| abstract  [Enumeration](http://docs.google.com/java/util/Enumeration.html) | [**children**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#children())()            Returns the children of the receiver as an Enumeration. |
| boolean | [**containsAttribute**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#containsAttribute(java.lang.Object,%20java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) name, [Object](http://docs.google.com/java/lang/Object.html) value)            Checks whether a given attribute name/value is defined. |
| boolean | [**containsAttributes**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#containsAttributes(javax.swing.text.AttributeSet))([AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attrs)            Checks whether the element contains all the attributes. |
| [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) | [**copyAttributes**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#copyAttributes())()            Copies a set of attributes. |
| void | [**dump**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#dump(java.io.PrintStream,%20int))([PrintStream](http://docs.google.com/java/io/PrintStream.html) psOut, int indentAmount)            Dumps a debugging representation of the element hierarchy. |
| abstract  boolean | [**getAllowsChildren**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getAllowsChildren())()            Returns true if the receiver allows children. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getAttribute**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getAttribute(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) attrName)            Gets the value of an attribute. |
| int | [**getAttributeCount**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getAttributeCount())()            Gets the number of attributes that are defined. |
| [Enumeration](http://docs.google.com/java/util/Enumeration.html)<?> | [**getAttributeNames**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getAttributeNames())()            Gets the names of all attributes. |
| [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) | [**getAttributes**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getAttributes())()            Gets the attributes for the element. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | [**getChildAt**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getChildAt(int))(int childIndex)            Returns the child TreeNode at index childIndex. |
| int | [**getChildCount**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getChildCount())()            Returns the number of children TreeNode's receiver contains. |
| [Document](http://docs.google.com/javax/swing/text/Document.html) | [**getDocument**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getDocument())()            Retrieves the underlying model. |
| abstract  [Element](http://docs.google.com/javax/swing/text/Element.html) | [**getElement**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getElement(int))(int index)            Gets a child element. |
| abstract  int | [**getElementCount**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getElementCount())()            Gets the number of children for the element. |
| abstract  int | [**getElementIndex**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getElementIndex(int))(int offset)            Gets the child element index closest to the given model offset. |
| abstract  int | [**getEndOffset**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getEndOffset())()            Gets the ending offset in the model for the element. |
| int | [**getIndex**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getIndex(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) node)            Returns the index of node in the receivers children. |
| [String](http://docs.google.com/java/lang/String.html) | [**getName**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getName())()            Gets the name of the element. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | [**getParent**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getParent())()            Returns the parent TreeNode of the receiver. |
| [Element](http://docs.google.com/javax/swing/text/Element.html) | [**getParentElement**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getParentElement())()            Gets the parent of the element. |
| [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) | [**getResolveParent**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getResolveParent())()            Gets the resolving parent. |
| abstract  int | [**getStartOffset**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getStartOffset())()            Gets the starting offset in the model for the element. |
| boolean | [**isDefined**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#isDefined(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) attrName)            Checks whether a given attribute is defined. |
| boolean | [**isEqual**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#isEqual(javax.swing.text.AttributeSet))([AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attr)            Checks whether two attribute sets are equal. |
| abstract  boolean | [**isLeaf**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#isLeaf())()            Checks whether the element is a leaf. |
| void | [**removeAttribute**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#removeAttribute(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) name)            Removes an attribute from the set. |
| void | [**removeAttributes**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#removeAttributes(javax.swing.text.AttributeSet))([AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attrs)            Removes a set of attributes for the element. |
| void | [**removeAttributes**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#removeAttributes(java.util.Enumeration))([Enumeration](http://docs.google.com/java/util/Enumeration.html)<?> names)            Removes a set of attributes for the element. |
| void | [**setResolveParent**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#setResolveParent(javax.swing.text.AttributeSet))([AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) parent)            Sets the resolving parent. |

| **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** |
| --- |

### AbstractDocument.AbstractElement

public **AbstractDocument.AbstractElement**([Element](http://docs.google.com/javax/swing/text/Element.html) parent,  
 [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) a)

Creates a new AbstractElement.

**Parameters:**parent - the parent elementa - the attributes for the element**Since:** 1.4

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

### dump

public void **dump**([PrintStream](http://docs.google.com/java/io/PrintStream.html) psOut,  
 int indentAmount)

Dumps a debugging representation of the element hierarchy.

**Parameters:**psOut - the output streamindentAmount - the indentation level >= 0

### getAttributeCount

public int **getAttributeCount**()

Gets the number of attributes that are defined.

**Specified by:**[getAttributeCount](http://docs.google.com/javax/swing/text/AttributeSet.html#getAttributeCount()) in interface [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) **Returns:**the number of attributes >= 0**See Also:**[AttributeSet.getAttributeCount()](http://docs.google.com/javax/swing/text/AttributeSet.html#getAttributeCount())

### isDefined

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

Checks whether a given attribute is defined.

**Specified by:**[isDefined](http://docs.google.com/javax/swing/text/AttributeSet.html#isDefined(java.lang.Object)) in interface [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) **Parameters:**attrName - the non-null attribute name **Returns:**true if the attribute is defined**See Also:**[AttributeSet.isDefined(java.lang.Object)](http://docs.google.com/javax/swing/text/AttributeSet.html#isDefined(java.lang.Object))

### isEqual

public boolean **isEqual**([AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attr)

Checks whether two attribute sets are equal.

**Specified by:**[isEqual](http://docs.google.com/javax/swing/text/AttributeSet.html#isEqual(javax.swing.text.AttributeSet)) in interface [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) **Parameters:**attr - the attribute set to check against **Returns:**true if the same**See Also:**[AttributeSet.isEqual(javax.swing.text.AttributeSet)](http://docs.google.com/javax/swing/text/AttributeSet.html#isEqual(javax.swing.text.AttributeSet))

### copyAttributes

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

Copies a set of attributes.

**Specified by:**[copyAttributes](http://docs.google.com/javax/swing/text/AttributeSet.html#copyAttributes()) in interface [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) **Returns:**the copy**See Also:**[AttributeSet.copyAttributes()](http://docs.google.com/javax/swing/text/AttributeSet.html#copyAttributes())

### getAttribute

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

Gets the value of an attribute.

**Specified by:**[getAttribute](http://docs.google.com/javax/swing/text/AttributeSet.html#getAttribute(java.lang.Object)) in interface [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) **Parameters:**attrName - the non-null attribute name **Returns:**the attribute value**See Also:**[AttributeSet.getAttribute(java.lang.Object)](http://docs.google.com/javax/swing/text/AttributeSet.html#getAttribute(java.lang.Object))

### getAttributeNames

public [Enumeration](http://docs.google.com/java/util/Enumeration.html)<?> **getAttributeNames**()

Gets the names of all attributes.

**Specified by:**[getAttributeNames](http://docs.google.com/javax/swing/text/AttributeSet.html#getAttributeNames()) in interface [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) **Returns:**the attribute names as an enumeration**See Also:**[AttributeSet.getAttributeNames()](http://docs.google.com/javax/swing/text/AttributeSet.html#getAttributeNames())

### containsAttribute

public boolean **containsAttribute**([Object](http://docs.google.com/java/lang/Object.html) name,  
 [Object](http://docs.google.com/java/lang/Object.html) value)

Checks whether a given attribute name/value is defined.

**Specified by:**[containsAttribute](http://docs.google.com/javax/swing/text/AttributeSet.html#containsAttribute(java.lang.Object,%20java.lang.Object)) in interface [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) **Parameters:**name - the non-null attribute namevalue - the attribute value **Returns:**true if the name/value is defined**See Also:**[AttributeSet.containsAttribute(java.lang.Object, java.lang.Object)](http://docs.google.com/javax/swing/text/AttributeSet.html#containsAttribute(java.lang.Object,%20java.lang.Object))

### containsAttributes

public boolean **containsAttributes**([AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attrs)

Checks whether the element contains all the attributes.

**Specified by:**[containsAttributes](http://docs.google.com/javax/swing/text/AttributeSet.html#containsAttributes(javax.swing.text.AttributeSet)) in interface [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) **Parameters:**attrs - the attributes to check **Returns:**true if the element contains all the attributes**See Also:**[AttributeSet.containsAttributes(javax.swing.text.AttributeSet)](http://docs.google.com/javax/swing/text/AttributeSet.html#containsAttributes(javax.swing.text.AttributeSet))

### getResolveParent

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

Gets the resolving parent. If not overridden, the resolving parent defaults to the parent element.

**Specified by:**[getResolveParent](http://docs.google.com/javax/swing/text/AttributeSet.html#getResolveParent()) in interface [AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) **Returns:**the attributes from the parent, null if none**See Also:**[AttributeSet.getResolveParent()](http://docs.google.com/javax/swing/text/AttributeSet.html#getResolveParent())

### addAttribute

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

Adds an attribute to the element.

**Specified by:**[addAttribute](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#addAttribute(java.lang.Object,%20java.lang.Object)) in interface [MutableAttributeSet](http://docs.google.com/javax/swing/text/MutableAttributeSet.html) **Parameters:**name - the non-null attribute namevalue - the attribute value**See Also:**[MutableAttributeSet.addAttribute(java.lang.Object, java.lang.Object)](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#addAttribute(java.lang.Object,%20java.lang.Object))

### addAttributes

public void **addAttributes**([AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attr)

Adds a set of attributes to the element.

**Specified by:**[addAttributes](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#addAttributes(javax.swing.text.AttributeSet)) in interface [MutableAttributeSet](http://docs.google.com/javax/swing/text/MutableAttributeSet.html) **Parameters:**attr - the attributes to add**See Also:**[MutableAttributeSet.addAttribute(java.lang.Object, java.lang.Object)](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#addAttribute(java.lang.Object,%20java.lang.Object))

### removeAttribute

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

Removes an attribute from the set.

**Specified by:**[removeAttribute](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#removeAttribute(java.lang.Object)) in interface [MutableAttributeSet](http://docs.google.com/javax/swing/text/MutableAttributeSet.html) **Parameters:**name - the non-null attribute name**See Also:**[MutableAttributeSet.removeAttribute(java.lang.Object)](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#removeAttribute(java.lang.Object))

### removeAttributes

public void **removeAttributes**([Enumeration](http://docs.google.com/java/util/Enumeration.html)<?> names)

Removes a set of attributes for the element.

**Specified by:**[removeAttributes](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#removeAttributes(java.util.Enumeration)) in interface [MutableAttributeSet](http://docs.google.com/javax/swing/text/MutableAttributeSet.html) **Parameters:**names - the attribute names**See Also:**[MutableAttributeSet.removeAttributes(java.util.Enumeration)](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#removeAttributes(java.util.Enumeration))

### removeAttributes

public void **removeAttributes**([AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) attrs)

Removes a set of attributes for the element.

**Specified by:**[removeAttributes](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#removeAttributes(javax.swing.text.AttributeSet)) in interface [MutableAttributeSet](http://docs.google.com/javax/swing/text/MutableAttributeSet.html) **Parameters:**attrs - the attributes**See Also:**[MutableAttributeSet.removeAttributes(java.util.Enumeration)](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#removeAttributes(java.util.Enumeration))

### setResolveParent

public void **setResolveParent**([AttributeSet](http://docs.google.com/javax/swing/text/AttributeSet.html) parent)

Sets the resolving parent.

**Specified by:**[setResolveParent](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#setResolveParent(javax.swing.text.AttributeSet)) in interface [MutableAttributeSet](http://docs.google.com/javax/swing/text/MutableAttributeSet.html) **Parameters:**parent - the parent, null if none**See Also:**[MutableAttributeSet.setResolveParent(javax.swing.text.AttributeSet)](http://docs.google.com/javax/swing/text/MutableAttributeSet.html#setResolveParent(javax.swing.text.AttributeSet))

### getDocument

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

Retrieves the underlying model.

**Specified by:**[getDocument](http://docs.google.com/javax/swing/text/Element.html#getDocument()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html) **Returns:**the model

### getParentElement

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

Gets the parent of the element.

**Specified by:**[getParentElement](http://docs.google.com/javax/swing/text/Element.html#getParentElement()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html) **Returns:**the parent

### getAttributes

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

Gets the attributes for the element.

**Specified by:**[getAttributes](http://docs.google.com/javax/swing/text/Element.html#getAttributes()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html) **Returns:**the attribute set

### getName

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

Gets the name of the element.

**Specified by:**[getName](http://docs.google.com/javax/swing/text/Element.html#getName()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html) **Returns:**the name, null if none

### getStartOffset

public abstract int **getStartOffset**()

Gets the starting offset in the model for the element.

**Specified by:**[getStartOffset](http://docs.google.com/javax/swing/text/Element.html#getStartOffset()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html) **Returns:**the offset >= 0**See Also:**[Document](http://docs.google.com/javax/swing/text/Document.html), [AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html)

### getEndOffset

public abstract int **getEndOffset**()

Gets the ending offset in the model for the element.

**Specified by:**[getEndOffset](http://docs.google.com/javax/swing/text/Element.html#getEndOffset()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html) **Returns:**the offset >= 0**See Also:**[Document](http://docs.google.com/javax/swing/text/Document.html), [AbstractDocument](http://docs.google.com/javax/swing/text/AbstractDocument.html)

### getElement

public abstract [Element](http://docs.google.com/javax/swing/text/Element.html) **getElement**(int index)

Gets a child element.

**Specified by:**[getElement](http://docs.google.com/javax/swing/text/Element.html#getElement(int)) in interface [Element](http://docs.google.com/javax/swing/text/Element.html) **Parameters:**index - the child index, >= 0 && < getElementCount() **Returns:**the child element

### getElementCount

public abstract int **getElementCount**()

Gets the number of children for the element.

**Specified by:**[getElementCount](http://docs.google.com/javax/swing/text/Element.html#getElementCount()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html) **Returns:**the number of children >= 0

### getElementIndex

public abstract int **getElementIndex**(int offset)

Gets the child element index closest to the given model offset.

**Specified by:**[getElementIndex](http://docs.google.com/javax/swing/text/Element.html#getElementIndex(int)) in interface [Element](http://docs.google.com/javax/swing/text/Element.html) **Parameters:**offset - the offset >= 0 **Returns:**the element index >= 0

### isLeaf

public abstract boolean **isLeaf**()

Checks whether the element is a leaf.

**Specified by:**[isLeaf](http://docs.google.com/javax/swing/text/Element.html#isLeaf()) in interface [Element](http://docs.google.com/javax/swing/text/Element.html)**Specified by:**[isLeaf](http://docs.google.com/javax/swing/tree/TreeNode.html#isLeaf()) in interface [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) **Returns:**true if a leaf

### getChildAt

public [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) **getChildAt**(int childIndex)

Returns the child TreeNode at index childIndex.

**Specified by:**[getChildAt](http://docs.google.com/javax/swing/tree/TreeNode.html#getChildAt(int)) in interface [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)

### getChildCount

public int **getChildCount**()

Returns the number of children TreeNode's receiver contains.

**Specified by:**[getChildCount](http://docs.google.com/javax/swing/tree/TreeNode.html#getChildCount()) in interface [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) **Returns:**the number of children TreeNodews's receiver contains

### getParent

public [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) **getParent**()

Returns the parent TreeNode of the receiver.

**Specified by:**[getParent](http://docs.google.com/javax/swing/tree/TreeNode.html#getParent()) in interface [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) **Returns:**the parent TreeNode of the receiver

### getIndex

public int **getIndex**([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) node)

Returns the index of node in the receivers children. If the receiver does not contain node, -1 will be returned.

**Specified by:**[getIndex](http://docs.google.com/javax/swing/tree/TreeNode.html#getIndex(javax.swing.tree.TreeNode)) in interface [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) **Parameters:**node - the location of interest **Returns:**the index of node in the receiver's children, or -1 if absent

### getAllowsChildren

public abstract boolean **getAllowsChildren**()

Returns true if the receiver allows children.

**Specified by:**[getAllowsChildren](http://docs.google.com/javax/swing/tree/TreeNode.html#getAllowsChildren()) in interface [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) **Returns:**true if the receiver allows children, otherwise false

### children

public abstract [Enumeration](http://docs.google.com/java/util/Enumeration.html) **children**()

Returns the children of the receiver as an Enumeration.

**Specified by:**[children](http://docs.google.com/javax/swing/tree/TreeNode.html#children()) in interface [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) **Returns:**the children of the receiver as an Enumeration

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/AbstractDocument.AbstractElement.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/AbstractDocument.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/text/AbstractDocument.AttributeContext.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/text/AbstractDocument.AbstractElement.html)    [**NO FRAMES**](http://docs.google.com/AbstractDocument.AbstractElement.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#1t3h5sf) | [METHOD](#4d34og8) | DETAIL: FIELD | [CONSTR](#17dp8vu) | [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).