| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/javax/swing/tree/TreeNode.html) | **Use** | [**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   NEXT | [**FRAMES**](http://docs.google.com/index.html?javax/swing/tree//class-useTreeNode.html)    [**NO FRAMES**](http://docs.google.com/TreeNode.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

**Uses of Interface**

**javax.swing.tree.TreeNode**

| Packages that use [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| [**javax.swing**](#3znysh7) | Provides a set of "lightweight" (all-Java language) components that, to the maximum degree possible, work the same on all platforms. |
| [**javax.swing.text**](#2et92p0) | Provides classes and interfaces that deal with editable and noneditable text components. |
| [**javax.swing.text.html**](#tyjcwt) | Provides the class HTMLEditorKit and supporting classes for creating HTML text editors. |
| [**javax.swing.tree**](#3dy6vkm) | Provides classes and interfaces for dealing with javax.swing.JTree. |

| Uses of [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) | |
| --- | --- |

| Classes in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) that implement [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| static class | [**JTree.DynamicUtilTreeNode**](http://docs.google.com/javax/swing/JTree.DynamicUtilTreeNode.html)            DynamicUtilTreeNode can wrap vectors/hashtables/arrays/strings and create the appropriate children tree nodes as necessary. |

| Methods in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) that return [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **JTree.DynamicUtilTreeNode.**[**getChildAt**](http://docs.google.com/javax/swing/JTree.DynamicUtilTreeNode.html#getChildAt(int))(int index)            Subclassed to load the children, if necessary. |

| Constructors in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) with parameters of type [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| [**JTree**](http://docs.google.com/javax/swing/JTree.html#JTree(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) root)            Returns a JTree with the specified TreeNode as its root, which displays the root node. |
| [**JTree**](http://docs.google.com/javax/swing/JTree.html#JTree(javax.swing.tree.TreeNode,%20boolean))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) root, boolean asksAllowsChildren)            Returns a JTree with the specified TreeNode as its root, which displays the root node and which decides whether a node is a leaf node in the specified manner. |

| Uses of [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) | |
| --- | --- |

| Classes in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) that implement [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| class | [**AbstractDocument.AbstractElement**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html)            Implements the abstract part of an element. |
| class | [**AbstractDocument.BranchElement**](http://docs.google.com/javax/swing/text/AbstractDocument.BranchElement.html)            Implements a composite element that contains other elements. |
| class | [**AbstractDocument.LeafElement**](http://docs.google.com/javax/swing/text/AbstractDocument.LeafElement.html)            Implements an element that directly represents content of some kind. |
| protected  class | [**DefaultStyledDocument.SectionElement**](http://docs.google.com/javax/swing/text/DefaultStyledDocument.SectionElement.html)            Default root element for a document... |

| Methods in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) that return [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **AbstractDocument.AbstractElement.**[**getChildAt**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getChildAt(int))(int childIndex)            Returns the child TreeNode at index childIndex. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **AbstractDocument.AbstractElement.**[**getParent**](http://docs.google.com/javax/swing/text/AbstractDocument.AbstractElement.html#getParent())()            Returns the parent TreeNode of the receiver. |

| Methods in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) with parameters of type [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| int | **AbstractDocument.AbstractElement.**[**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. |

| Uses of [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) in [javax.swing.text.html](http://docs.google.com/javax/swing/text/html/package-summary.html) | |
| --- | --- |

| Classes in [javax.swing.text.html](http://docs.google.com/javax/swing/text/html/package-summary.html) that implement [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| class | [**HTMLDocument.BlockElement**](http://docs.google.com/javax/swing/text/html/HTMLDocument.BlockElement.html)            An element that represents a structural *block* of HTML. |
| class | [**HTMLDocument.RunElement**](http://docs.google.com/javax/swing/text/html/HTMLDocument.RunElement.html)            An element that represents a chunk of text that has a set of HTML character level attributes assigned to it. |

| Uses of [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) | |
| --- | --- |

| Subinterfaces of [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) | |
| --- | --- |
| interface | [**MutableTreeNode**](http://docs.google.com/javax/swing/tree/MutableTreeNode.html)            Defines the requirements for a tree node object that can change -- by adding or removing child nodes, or by changing the contents of a user object stored in the node. |

| Classes in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) that implement [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| class | [**DefaultMutableTreeNode**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html)            A DefaultMutableTreeNode is a general-purpose node in a tree data structure. |

| Fields in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) declared as [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| protected  [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultTreeModel.**[**root**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#root)            Root of the tree. |

| Fields in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) with type parameters of type [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| static [Enumeration](http://docs.google.com/java/util/Enumeration.html)<[TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)> | **DefaultMutableTreeNode.**[**EMPTY\_ENUMERATION**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#EMPTY_ENUMERATION)            An enumeration that is always empty. |

| Methods in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) that return [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultMutableTreeNode.**[**getChildAfter**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getChildAfter(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aChild)            Returns the child in this node's child array that immediately follows aChild, which must be a child of this node. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultMutableTreeNode.**[**getChildAt**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getChildAt(int))(int index)            Returns the child at the specified index in this node's child array. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **TreeNode.**[**getChildAt**](http://docs.google.com/javax/swing/tree/TreeNode.html#getChildAt(int))(int childIndex)            Returns the child TreeNode at index childIndex. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultMutableTreeNode.**[**getChildBefore**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getChildBefore(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aChild)            Returns the child in this node's child array that immediately precedes aChild, which must be a child of this node. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultMutableTreeNode.**[**getFirstChild**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getFirstChild())()            Returns this node's first child. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultMutableTreeNode.**[**getLastChild**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getLastChild())()            Returns this node's last child. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultMutableTreeNode.**[**getParent**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getParent())()            Returns this node's parent or null if this node has no parent. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **TreeNode.**[**getParent**](http://docs.google.com/javax/swing/tree/TreeNode.html#getParent())()            Returns the parent TreeNode of the receiver. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)[] | **DefaultMutableTreeNode.**[**getPath**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getPath())()            Returns the path from the root, to get to this node. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)[] | **DefaultTreeModel.**[**getPathToRoot**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#getPathToRoot(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aNode)            Builds the parents of node up to and including the root node, where the original node is the last element in the returned array. |
| protected  [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)[] | **DefaultTreeModel.**[**getPathToRoot**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#getPathToRoot(javax.swing.tree.TreeNode,%20int))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aNode, int depth)            Builds the parents of node up to and including the root node, where the original node is the last element in the returned array. |
| protected  [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)[] | **DefaultMutableTreeNode.**[**getPathToRoot**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getPathToRoot(javax.swing.tree.TreeNode,%20int))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aNode, int depth)            Builds the parents of node up to and including the root node, where the original node is the last element in the returned array. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultMutableTreeNode.**[**getRoot**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getRoot())()            Returns the root of the tree that contains this node. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultMutableTreeNode.**[**getSharedAncestor**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getSharedAncestor(javax.swing.tree.DefaultMutableTreeNode))([DefaultMutableTreeNode](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html) aNode)            Returns the nearest common ancestor to this node and aNode. |

| Methods in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) with parameters of type [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultMutableTreeNode.**[**getChildAfter**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getChildAfter(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aChild)            Returns the child in this node's child array that immediately follows aChild, which must be a child of this node. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | **DefaultMutableTreeNode.**[**getChildBefore**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getChildBefore(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aChild)            Returns the child in this node's child array that immediately precedes aChild, which must be a child of this node. |
| int | **DefaultMutableTreeNode.**[**getIndex**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getIndex(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aChild)            Returns the index of the specified child in this node's child array. |
| int | **TreeNode.**[**getIndex**](http://docs.google.com/javax/swing/tree/TreeNode.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. |
| [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)[] | **DefaultTreeModel.**[**getPathToRoot**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#getPathToRoot(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aNode)            Builds the parents of node up to and including the root node, where the original node is the last element in the returned array. |
| protected  [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)[] | **DefaultTreeModel.**[**getPathToRoot**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#getPathToRoot(javax.swing.tree.TreeNode,%20int))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aNode, int depth)            Builds the parents of node up to and including the root node, where the original node is the last element in the returned array. |
| protected  [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html)[] | **DefaultMutableTreeNode.**[**getPathToRoot**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#getPathToRoot(javax.swing.tree.TreeNode,%20int))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aNode, int depth)            Builds the parents of node up to and including the root node, where the original node is the last element in the returned array. |
| boolean | **DefaultMutableTreeNode.**[**isNodeAncestor**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#isNodeAncestor(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) anotherNode)            Returns true if anotherNode is an ancestor of this node -- if it is this node, this node's parent, or an ancestor of this node's parent. |
| boolean | **DefaultMutableTreeNode.**[**isNodeChild**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#isNodeChild(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) aNode)            Returns true if aNode is a child of this node. |
| boolean | **DefaultMutableTreeNode.**[**isNodeSibling**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#isNodeSibling(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) anotherNode)            Returns true if anotherNode is a sibling of (has the same parent as) this node. |
| void | **DefaultTreeModel.**[**nodeChanged**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#nodeChanged(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) node)            Invoke this method after you've changed how node is to be represented in the tree. |
| void | **DefaultTreeModel.**[**nodesChanged**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#nodesChanged(javax.swing.tree.TreeNode,%20int%5B%5D))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) node, int[] childIndices)            Invoke this method after you've changed how the children identified by childIndicies are to be represented in the tree. |
| void | **DefaultTreeModel.**[**nodeStructureChanged**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#nodeStructureChanged(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) node)            Invoke this method if you've totally changed the children of node and its childrens children... |
| void | **DefaultTreeModel.**[**nodesWereInserted**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#nodesWereInserted(javax.swing.tree.TreeNode,%20int%5B%5D))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) node, int[] childIndices)            Invoke this method after you've inserted some TreeNodes into node. |
| void | **DefaultTreeModel.**[**nodesWereRemoved**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#nodesWereRemoved(javax.swing.tree.TreeNode,%20int%5B%5D,%20java.lang.Object%5B%5D))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) node, int[] childIndices, [Object](http://docs.google.com/java/lang/Object.html)[] removedChildren)            Invoke this method after you've removed some TreeNodes from node. |
| [Enumeration](http://docs.google.com/java/util/Enumeration.html) | **DefaultMutableTreeNode.**[**pathFromAncestorEnumeration**](http://docs.google.com/javax/swing/tree/DefaultMutableTreeNode.html#pathFromAncestorEnumeration(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) ancestor)            Creates and returns an enumeration that follows the path from ancestor to this node. |
| void | **DefaultTreeModel.**[**reload**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#reload(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) node)            Invoke this method if you've modified the TreeNodes upon which this model depends. |
| void | **DefaultTreeModel.**[**setRoot**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#setRoot(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) root)            Sets the root to root. |

| Constructors in [javax.swing.tree](http://docs.google.com/javax/swing/tree/package-summary.html) with parameters of type [TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) | |
| --- | --- |
| [**DefaultTreeModel**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#DefaultTreeModel(javax.swing.tree.TreeNode))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) root)            Creates a tree in which any node can have children. |
| [**DefaultTreeModel**](http://docs.google.com/javax/swing/tree/DefaultTreeModel.html#DefaultTreeModel(javax.swing.tree.TreeNode,%20boolean))([TreeNode](http://docs.google.com/javax/swing/tree/TreeNode.html) root, boolean asksAllowsChildren)            Creates a tree specifying whether any node can have children, or whether only certain nodes can have children. |

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/javax/swing/tree/TreeNode.html) | **Use** | [**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   NEXT | [**FRAMES**](http://docs.google.com/index.html?javax/swing/tree//class-useTreeNode.html)    [**NO FRAMES**](http://docs.google.com/TreeNode.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[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).