| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/TreePath.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/tree/TreeNode.html)   [**NEXT CLASS**](http://docs.google.com/javax/swing/tree/TreeSelectionModel.html) | [**FRAMES**](http://docs.google.com/index.html?javax/swing/tree/TreePath.html)    [**NO FRAMES**](http://docs.google.com/TreePath.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#3dy6vkm) | [METHOD](#26in1rg) |

## **javax.swing.tree**

Class TreePath

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 **javax.swing.tree.TreePath**

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

public class **TreePath**extends [Object](http://docs.google.com/java/lang/Object.html)implements [Serializable](http://docs.google.com/java/io/Serializable.html)

Represents a path to a node. A TreePath is an array of Objects that are vended from a TreeModel. The elements of the array are ordered such that the root is always the first element (index 0) of the array. TreePath is Serializable, but if any components of the path are not serializable, it will not be written out.

For further information and examples of using tree paths, see [How to Use Trees](http://java.sun.com/docs/books/tutorial/uiswing/components/tree.html) in *The Java Tutorial.*

**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).

| **Constructor Summary** | |
| --- | --- |
| protected | [**TreePath**](http://docs.google.com/javax/swing/tree/TreePath.html#TreePath())()            Primarily provided for subclasses that represent paths in a different manner. |
|  | [**TreePath**](http://docs.google.com/javax/swing/tree/TreePath.html#TreePath(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) singlePath)            Constructs a TreePath containing only a single element. |
|  | [**TreePath**](http://docs.google.com/javax/swing/tree/TreePath.html#TreePath(java.lang.Object%5B%5D))([Object](http://docs.google.com/java/lang/Object.html)[] path)            Constructs a path from an array of Objects, uniquely identifying the path from the root of the tree to a specific node, as returned by the tree's data model. |
| protected | [**TreePath**](http://docs.google.com/javax/swing/tree/TreePath.html#TreePath(java.lang.Object%5B%5D,%20int))([Object](http://docs.google.com/java/lang/Object.html)[] path, int length)            Constructs a new TreePath with the identified path components of length length. |
| protected | [**TreePath**](http://docs.google.com/javax/swing/tree/TreePath.html#TreePath(javax.swing.tree.TreePath,%20java.lang.Object))([TreePath](http://docs.google.com/javax/swing/tree/TreePath.html) parent, [Object](http://docs.google.com/java/lang/Object.html) lastElement)            Constructs a new TreePath, which is the path identified by parent ending in lastElement. |

| **Method Summary** | |
| --- | --- |
| boolean | [**equals**](http://docs.google.com/javax/swing/tree/TreePath.html#equals(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) o)            Tests two TreePaths for equality by checking each element of the paths for equality. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getLastPathComponent**](http://docs.google.com/javax/swing/tree/TreePath.html#getLastPathComponent())()            Returns the last component of this path. |
| [TreePath](http://docs.google.com/javax/swing/tree/TreePath.html) | [**getParentPath**](http://docs.google.com/javax/swing/tree/TreePath.html#getParentPath())()            Returns a path containing all the elements of this object, except the last path component. |
| [Object](http://docs.google.com/java/lang/Object.html)[] | [**getPath**](http://docs.google.com/javax/swing/tree/TreePath.html#getPath())()            Returns an ordered array of Objects containing the components of this TreePath. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getPathComponent**](http://docs.google.com/javax/swing/tree/TreePath.html#getPathComponent(int))(int element)            Returns the path component at the specified index. |
| int | [**getPathCount**](http://docs.google.com/javax/swing/tree/TreePath.html#getPathCount())()            Returns the number of elements in the path. |
| int | [**hashCode**](http://docs.google.com/javax/swing/tree/TreePath.html#hashCode())()            Returns the hashCode for the object. |
| boolean | [**isDescendant**](http://docs.google.com/javax/swing/tree/TreePath.html#isDescendant(javax.swing.tree.TreePath))([TreePath](http://docs.google.com/javax/swing/tree/TreePath.html) aTreePath)            Returns true if aTreePath is a descendant of this TreePath. |
| [TreePath](http://docs.google.com/javax/swing/tree/TreePath.html) | [**pathByAddingChild**](http://docs.google.com/javax/swing/tree/TreePath.html#pathByAddingChild(java.lang.Object))([Object](http://docs.google.com/java/lang/Object.html) child)            Returns a new path containing all the elements of this object plus child. |
| [String](http://docs.google.com/java/lang/String.html) | [**toString**](http://docs.google.com/javax/swing/tree/TreePath.html#toString())()            Returns a string that displays and identifies this object's properties. |

| **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()), [finalize](http://docs.google.com/java/lang/Object.html#finalize()), [getClass](http://docs.google.com/java/lang/Object.html#getClass()), [notify](http://docs.google.com/java/lang/Object.html#notify()), [notifyAll](http://docs.google.com/java/lang/Object.html#notifyAll()), [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** |
| --- |

### TreePath

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

Constructs a path from an array of Objects, uniquely identifying the path from the root of the tree to a specific node, as returned by the tree's data model.

The model is free to return an array of any Objects it needs to represent the path. The DefaultTreeModel returns an array of TreeNode objects. The first TreeNode in the path is the root of the tree, the last TreeNode is the node identified by the path.

**Parameters:**path - an array of Objects representing the path to a node

### TreePath

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

Constructs a TreePath containing only a single element. This is usually used to construct a TreePath for the the root of the TreeModel.

**Parameters:**singlePath - an Object representing the path to a node**See Also:**[TreePath(Object[])](http://docs.google.com/javax/swing/tree/TreePath.html#TreePath(java.lang.Object%5B%5D))

### TreePath

protected **TreePath**([TreePath](http://docs.google.com/javax/swing/tree/TreePath.html) parent,  
 [Object](http://docs.google.com/java/lang/Object.html) lastElement)

Constructs a new TreePath, which is the path identified by parent ending in lastElement.

### TreePath

protected **TreePath**([Object](http://docs.google.com/java/lang/Object.html)[] path,  
 int length)

Constructs a new TreePath with the identified path components of length length.

### TreePath

protected **TreePath**()

Primarily provided for subclasses that represent paths in a different manner. If a subclass uses this constructor, it should also override the getPath, getPathCount, and getPathComponent methods, and possibly the equals method.

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

### getPath

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

Returns an ordered array of Objects containing the components of this TreePath. The first element (index 0) is the root.

**Returns:**an array of Objects representing the TreePath**See Also:**[TreePath(Object[])](http://docs.google.com/javax/swing/tree/TreePath.html#TreePath(java.lang.Object%5B%5D))

### getLastPathComponent

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

Returns the last component of this path. For a path returned by DefaultTreeModel this will return an instance of TreeNode.

**Returns:**the Object at the end of the path**See Also:**[TreePath(Object[])](http://docs.google.com/javax/swing/tree/TreePath.html#TreePath(java.lang.Object%5B%5D))

### getPathCount

public int **getPathCount**()

Returns the number of elements in the path.

**Returns:**an int giving a count of items the path

### getPathComponent

public [Object](http://docs.google.com/java/lang/Object.html) **getPathComponent**(int element)

Returns the path component at the specified index.

**Parameters:**element - an int specifying an element in the path, where 0 is the first element in the path **Returns:**the Object at that index location **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if the index is beyond the length of the path**See Also:**[TreePath(Object[])](http://docs.google.com/javax/swing/tree/TreePath.html#TreePath(java.lang.Object%5B%5D))

### equals

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

Tests two TreePaths for equality by checking each element of the paths for equality. Two paths are considered equal if they are of the same length, and contain the same elements (.equals).

**Overrides:**[equals](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)) in class [Object](http://docs.google.com/java/lang/Object.html) **Parameters:**o - the Object to compare **Returns:**true if this object is the same as the obj argument; false otherwise.**See Also:**[Object.hashCode()](http://docs.google.com/java/lang/Object.html#hashCode()), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### hashCode

public int **hashCode**()

Returns the hashCode for the object. The hash code of a TreePath is defined to be the hash code of the last component in the path.

**Overrides:**[hashCode](http://docs.google.com/java/lang/Object.html#hashCode()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**the hashCode for the object**See Also:**[Object.equals(java.lang.Object)](http://docs.google.com/java/lang/Object.html#equals(java.lang.Object)), [Hashtable](http://docs.google.com/java/util/Hashtable.html)

### isDescendant

public boolean **isDescendant**([TreePath](http://docs.google.com/javax/swing/tree/TreePath.html) aTreePath)

Returns true if aTreePath is a descendant of this TreePath. A TreePath P1 is a descendant of a TreePath P2 if P1 contains all of the components that make up P2's path. For example, if this object has the path [a, b], and aTreePath has the path [a, b, c], then aTreePath is a descendant of this object. However, if aTreePath has the path [a], then it is not a descendant of this object. By this definition a TreePath is always considered a descendant of itself. That is, aTreePath.isDescendant(aTreePath) returns true.

**Returns:**true if aTreePath is a descendant of this path

### pathByAddingChild

public [TreePath](http://docs.google.com/javax/swing/tree/TreePath.html) **pathByAddingChild**([Object](http://docs.google.com/java/lang/Object.html) child)

Returns a new path containing all the elements of this object plus child. child will be the last element of the newly created TreePath. This will throw a NullPointerException if child is null.

### getParentPath

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

Returns a path containing all the elements of this object, except the last path component.

### toString

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

Returns a string that displays and identifies this object's properties.

**Overrides:**[toString](http://docs.google.com/java/lang/Object.html#toString()) in class [Object](http://docs.google.com/java/lang/Object.html) **Returns:**a String representation of this object

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