| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/CheckboxMenuItem.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/java/awt/CheckboxGroup.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/CheckboxMenuItem.AccessibleAWTCheckboxMenuItem.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/CheckboxMenuItem.html)    [**NO FRAMES**](http://docs.google.com/CheckboxMenuItem.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | FIELD | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: FIELD | [CONSTR](#3rdcrjn) | [METHOD](#1ksv4uv) |

## **java.awt**

Class CheckboxMenuItem

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.awt.MenuComponent](http://docs.google.com/java/awt/MenuComponent.html)  
 [java.awt.MenuItem](http://docs.google.com/java/awt/MenuItem.html)  
 **java.awt.CheckboxMenuItem**

**All Implemented Interfaces:** [ItemSelectable](http://docs.google.com/java/awt/ItemSelectable.html), [Serializable](http://docs.google.com/java/io/Serializable.html), [Accessible](http://docs.google.com/javax/accessibility/Accessible.html)

public class **CheckboxMenuItem**extends [MenuItem](http://docs.google.com/java/awt/MenuItem.html)implements [ItemSelectable](http://docs.google.com/java/awt/ItemSelectable.html), [Accessible](http://docs.google.com/javax/accessibility/Accessible.html)

This class represents a check box that can be included in a menu. Selecting the check box in the menu changes its state from "on" to "off" or from "off" to "on."

The following picture depicts a menu which contains an instance of CheckBoxMenuItem:



The item labeled Check shows a check box menu item in its "off" state.

When a check box menu item is selected, AWT sends an item event to the item. Since the event is an instance of ItemEvent, the processEvent method examines the event and passes it along to processItemEvent. The latter method redirects the event to any ItemListener objects that have registered an interest in item events generated by this menu item.

**Since:** JDK1.0 **See Also:**[ItemEvent](http://docs.google.com/java/awt/event/ItemEvent.html), [ItemListener](http://docs.google.com/java/awt/event/ItemListener.html), [Serialized Form](http://docs.google.com/serialized-form.html#java.awt.CheckboxMenuItem)

| **Nested Class Summary** | |
| --- | --- |
| protected  class | [**CheckboxMenuItem.AccessibleAWTCheckboxMenuItem**](http://docs.google.com/java/awt/CheckboxMenuItem.AccessibleAWTCheckboxMenuItem.html)            Inner class of CheckboxMenuItem used to provide default support for accessibility. |

| **Nested classes/interfaces inherited from class java.awt.**[**MenuItem**](http://docs.google.com/java/awt/MenuItem.html) |
| --- |
| [MenuItem.AccessibleAWTMenuItem](http://docs.google.com/java/awt/MenuItem.AccessibleAWTMenuItem.html) |

| **Nested classes/interfaces inherited from class java.awt.**[**MenuComponent**](http://docs.google.com/java/awt/MenuComponent.html) |
| --- |
| [MenuComponent.AccessibleAWTMenuComponent](http://docs.google.com/java/awt/MenuComponent.AccessibleAWTMenuComponent.html) |

| **Constructor Summary** | |
| --- | --- |
| [**CheckboxMenuItem**](http://docs.google.com/java/awt/CheckboxMenuItem.html#CheckboxMenuItem())()            Create a check box menu item with an empty label. |
| [**CheckboxMenuItem**](http://docs.google.com/java/awt/CheckboxMenuItem.html#CheckboxMenuItem(java.lang.String))([String](http://docs.google.com/java/lang/String.html) label)            Create a check box menu item with the specified label. |
| [**CheckboxMenuItem**](http://docs.google.com/java/awt/CheckboxMenuItem.html#CheckboxMenuItem(java.lang.String,%20boolean))([String](http://docs.google.com/java/lang/String.html) label, boolean state)            Create a check box menu item with the specified label and state. |

| **Method Summary** | |
| --- | --- |
| void | [**addItemListener**](http://docs.google.com/java/awt/CheckboxMenuItem.html#addItemListener(java.awt.event.ItemListener))([ItemListener](http://docs.google.com/java/awt/event/ItemListener.html) l)            Adds the specified item listener to receive item events from this check box menu item. |
| void | [**addNotify**](http://docs.google.com/java/awt/CheckboxMenuItem.html#addNotify())()            Creates the peer of the checkbox item. |
| [AccessibleContext](http://docs.google.com/javax/accessibility/AccessibleContext.html) | [**getAccessibleContext**](http://docs.google.com/java/awt/CheckboxMenuItem.html#getAccessibleContext())()            Gets the AccessibleContext associated with this CheckboxMenuItem. |
| [ItemListener](http://docs.google.com/java/awt/event/ItemListener.html)[] | [**getItemListeners**](http://docs.google.com/java/awt/CheckboxMenuItem.html#getItemListeners())()            Returns an array of all the item listeners registered on this checkbox menuitem. |
| | <T extends [EventListener](http://docs.google.com/java/util/EventListener.html)>  T[] | | --- | | [**getListeners**](http://docs.google.com/java/awt/CheckboxMenuItem.html#getListeners(java.lang.Class))([Class](http://docs.google.com/java/lang/Class.html)<T> listenerType)            Returns an array of all the objects currently registered as *Foo*Listeners upon this CheckboxMenuItem. |
| [Object](http://docs.google.com/java/lang/Object.html)[] | [**getSelectedObjects**](http://docs.google.com/java/awt/CheckboxMenuItem.html#getSelectedObjects())()            Returns the an array (length 1) containing the checkbox menu item label or null if the checkbox is not selected. |
| boolean | [**getState**](http://docs.google.com/java/awt/CheckboxMenuItem.html#getState())()            Determines whether the state of this check box menu item is "on" or "off." |
| [String](http://docs.google.com/java/lang/String.html) | [**paramString**](http://docs.google.com/java/awt/CheckboxMenuItem.html#paramString())()            Returns a string representing the state of this CheckBoxMenuItem. |
| protected  void | [**processEvent**](http://docs.google.com/java/awt/CheckboxMenuItem.html#processEvent(java.awt.AWTEvent))([AWTEvent](http://docs.google.com/java/awt/AWTEvent.html) e)            Processes events on this check box menu item. |
| protected  void | [**processItemEvent**](http://docs.google.com/java/awt/CheckboxMenuItem.html#processItemEvent(java.awt.event.ItemEvent))([ItemEvent](http://docs.google.com/java/awt/event/ItemEvent.html) e)            Processes item events occurring on this check box menu item by dispatching them to any registered ItemListener objects. |
| void | [**removeItemListener**](http://docs.google.com/java/awt/CheckboxMenuItem.html#removeItemListener(java.awt.event.ItemListener))([ItemListener](http://docs.google.com/java/awt/event/ItemListener.html) l)            Removes the specified item listener so that it no longer receives item events from this check box menu item. |
| void | [**setState**](http://docs.google.com/java/awt/CheckboxMenuItem.html#setState(boolean))(boolean b)            Sets this check box menu item to the specifed state. |

| **Methods inherited from class java.awt.**[**MenuItem**](http://docs.google.com/java/awt/MenuItem.html) |
| --- |
| [addActionListener](http://docs.google.com/java/awt/MenuItem.html#addActionListener(java.awt.event.ActionListener)), [deleteShortcut](http://docs.google.com/java/awt/MenuItem.html#deleteShortcut()), [disable](http://docs.google.com/java/awt/MenuItem.html#disable()), [disableEvents](http://docs.google.com/java/awt/MenuItem.html#disableEvents(long)), [enable](http://docs.google.com/java/awt/MenuItem.html#enable()), [enable](http://docs.google.com/java/awt/MenuItem.html#enable(boolean)), [enableEvents](http://docs.google.com/java/awt/MenuItem.html#enableEvents(long)), [getActionCommand](http://docs.google.com/java/awt/MenuItem.html#getActionCommand()), [getActionListeners](http://docs.google.com/java/awt/MenuItem.html#getActionListeners()), [getLabel](http://docs.google.com/java/awt/MenuItem.html#getLabel()), [getShortcut](http://docs.google.com/java/awt/MenuItem.html#getShortcut()), [isEnabled](http://docs.google.com/java/awt/MenuItem.html#isEnabled()), [processActionEvent](http://docs.google.com/java/awt/MenuItem.html#processActionEvent(java.awt.event.ActionEvent)), [removeActionListener](http://docs.google.com/java/awt/MenuItem.html#removeActionListener(java.awt.event.ActionListener)), [setActionCommand](http://docs.google.com/java/awt/MenuItem.html#setActionCommand(java.lang.String)), [setEnabled](http://docs.google.com/java/awt/MenuItem.html#setEnabled(boolean)), [setLabel](http://docs.google.com/java/awt/MenuItem.html#setLabel(java.lang.String)), [setShortcut](http://docs.google.com/java/awt/MenuItem.html#setShortcut(java.awt.MenuShortcut)) |

| **Methods inherited from class java.awt.**[**MenuComponent**](http://docs.google.com/java/awt/MenuComponent.html) |
| --- |
| [dispatchEvent](http://docs.google.com/java/awt/MenuComponent.html#dispatchEvent(java.awt.AWTEvent)), [getFont](http://docs.google.com/java/awt/MenuComponent.html#getFont()), [getName](http://docs.google.com/java/awt/MenuComponent.html#getName()), [getParent](http://docs.google.com/java/awt/MenuComponent.html#getParent()), [getPeer](http://docs.google.com/java/awt/MenuComponent.html#getPeer()), [getTreeLock](http://docs.google.com/java/awt/MenuComponent.html#getTreeLock()), [postEvent](http://docs.google.com/java/awt/MenuComponent.html#postEvent(java.awt.Event)), [removeNotify](http://docs.google.com/java/awt/MenuComponent.html#removeNotify()), [setFont](http://docs.google.com/java/awt/MenuComponent.html#setFont(java.awt.Font)), [setName](http://docs.google.com/java/awt/MenuComponent.html#setName(java.lang.String)), [toString](http://docs.google.com/java/awt/MenuComponent.html#toString()) |

| **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()), [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** |
| --- |

### CheckboxMenuItem

public **CheckboxMenuItem**()  
 throws [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html)

Create a check box menu item with an empty label. The item's state is initially set to "off."

**Throws:** [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html) - if GraphicsEnvironment.isHeadless() returns true**Since:** JDK1.1 **See Also:**[GraphicsEnvironment.isHeadless()](http://docs.google.com/java/awt/GraphicsEnvironment.html#isHeadless())

### CheckboxMenuItem

public **CheckboxMenuItem**([String](http://docs.google.com/java/lang/String.html) label)  
 throws [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html)

Create a check box menu item with the specified label. The item's state is initially set to "off."

**Parameters:**label - a string label for the check box menu item, or null for an unlabeled menu item. **Throws:** [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html) - if GraphicsEnvironment.isHeadless() returns true**See Also:**[GraphicsEnvironment.isHeadless()](http://docs.google.com/java/awt/GraphicsEnvironment.html#isHeadless())

### CheckboxMenuItem

public **CheckboxMenuItem**([String](http://docs.google.com/java/lang/String.html) label,  
 boolean state)  
 throws [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html)

Create a check box menu item with the specified label and state.

**Parameters:**label - a string label for the check box menu item, or null for an unlabeled menu item.state - the initial state of the menu item, where true indicates "on" and false indicates "off." **Throws:** [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html) - if GraphicsEnvironment.isHeadless() returns true**Since:** JDK1.1 **See Also:**[GraphicsEnvironment.isHeadless()](http://docs.google.com/java/awt/GraphicsEnvironment.html#isHeadless())

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

### addNotify

public void **addNotify**()

Creates the peer of the checkbox item. This peer allows us to change the look of the checkbox item without changing its functionality. Most applications do not call this method directly.

**Overrides:**[addNotify](http://docs.google.com/java/awt/MenuItem.html#addNotify()) in class [MenuItem](http://docs.google.com/java/awt/MenuItem.html) **See Also:**[Toolkit.createCheckboxMenuItem(java.awt.CheckboxMenuItem)](http://docs.google.com/java/awt/Toolkit.html#createCheckboxMenuItem(java.awt.CheckboxMenuItem)), [Component.getToolkit()](http://docs.google.com/java/awt/Component.html#getToolkit())

### getState

public boolean **getState**()

Determines whether the state of this check box menu item is "on" or "off."

**Returns:**the state of this check box menu item, where true indicates "on" and false indicates "off"**See Also:**[setState(boolean)](http://docs.google.com/java/awt/CheckboxMenuItem.html#setState(boolean))

### setState

public void **setState**(boolean b)

Sets this check box menu item to the specifed state. The boolean value true indicates "on" while false indicates "off."

Note that this method should be primarily used to initialize the state of the check box menu item. Programmatically setting the state of the check box menu item will *not* trigger an ItemEvent. The only way to trigger an ItemEvent is by user interaction.

**Parameters:**b - true if the check box menu item is on, otherwise false**See Also:**[getState()](http://docs.google.com/java/awt/CheckboxMenuItem.html#getState())

### getSelectedObjects

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

Returns the an array (length 1) containing the checkbox menu item label or null if the checkbox is not selected.

**Specified by:**[getSelectedObjects](http://docs.google.com/java/awt/ItemSelectable.html#getSelectedObjects()) in interface [ItemSelectable](http://docs.google.com/java/awt/ItemSelectable.html) **See Also:**[ItemSelectable](http://docs.google.com/java/awt/ItemSelectable.html)

### addItemListener

public void **addItemListener**([ItemListener](http://docs.google.com/java/awt/event/ItemListener.html) l)

Adds the specified item listener to receive item events from this check box menu item. Item events are sent in response to user actions, but not in response to calls to setState(). If l is null, no exception is thrown and no action is performed.

Refer to [AWT Threading Issues](http://docs.google.com/doc-files/AWTThreadIssues.html#ListenersThreads) for details on AWT's threading model.

**Specified by:**[addItemListener](http://docs.google.com/java/awt/ItemSelectable.html#addItemListener(java.awt.event.ItemListener)) in interface [ItemSelectable](http://docs.google.com/java/awt/ItemSelectable.html) **Parameters:**l - the item listener**Since:** JDK1.1 **See Also:**[removeItemListener(java.awt.event.ItemListener)](http://docs.google.com/java/awt/CheckboxMenuItem.html#removeItemListener(java.awt.event.ItemListener)), [getItemListeners()](http://docs.google.com/java/awt/CheckboxMenuItem.html#getItemListeners()), [setState(boolean)](http://docs.google.com/java/awt/CheckboxMenuItem.html#setState(boolean)), [ItemEvent](http://docs.google.com/java/awt/event/ItemEvent.html), [ItemListener](http://docs.google.com/java/awt/event/ItemListener.html)

### removeItemListener

public void **removeItemListener**([ItemListener](http://docs.google.com/java/awt/event/ItemListener.html) l)

Removes the specified item listener so that it no longer receives item events from this check box menu item. If l is null, no exception is thrown and no action is performed.

Refer to [AWT Threading Issues](http://docs.google.com/doc-files/AWTThreadIssues.html#ListenersThreads) for details on AWT's threading model.

**Specified by:**[removeItemListener](http://docs.google.com/java/awt/ItemSelectable.html#removeItemListener(java.awt.event.ItemListener)) in interface [ItemSelectable](http://docs.google.com/java/awt/ItemSelectable.html) **Parameters:**l - the item listener**Since:** JDK1.1 **See Also:**[addItemListener(java.awt.event.ItemListener)](http://docs.google.com/java/awt/CheckboxMenuItem.html#addItemListener(java.awt.event.ItemListener)), [getItemListeners()](http://docs.google.com/java/awt/CheckboxMenuItem.html#getItemListeners()), [ItemEvent](http://docs.google.com/java/awt/event/ItemEvent.html), [ItemListener](http://docs.google.com/java/awt/event/ItemListener.html)

### getItemListeners

public [ItemListener](http://docs.google.com/java/awt/event/ItemListener.html)[] **getItemListeners**()

Returns an array of all the item listeners registered on this checkbox menuitem.

**Returns:**all of this checkbox menuitem's ItemListeners or an empty array if no item listeners are currently registered**Since:** 1.4 **See Also:**[addItemListener(java.awt.event.ItemListener)](http://docs.google.com/java/awt/CheckboxMenuItem.html#addItemListener(java.awt.event.ItemListener)), [removeItemListener(java.awt.event.ItemListener)](http://docs.google.com/java/awt/CheckboxMenuItem.html#removeItemListener(java.awt.event.ItemListener)), [ItemEvent](http://docs.google.com/java/awt/event/ItemEvent.html), [ItemListener](http://docs.google.com/java/awt/event/ItemListener.html)

### getListeners

public <T extends [EventListener](http://docs.google.com/java/util/EventListener.html)> T[] **getListeners**([Class](http://docs.google.com/java/lang/Class.html)<T> listenerType)

Returns an array of all the objects currently registered as *Foo*Listeners upon this CheckboxMenuItem. *Foo*Listeners are registered using the add*Foo*Listener method.

You can specify the listenerType argument with a class literal, such as *Foo*Listener.class. For example, you can query a CheckboxMenuItem c for its item listeners with the following code:

ItemListener[] ils = (ItemListener[])(c.getListeners(ItemListener.class));

If no such listeners exist, this method returns an empty array.

**Overrides:**[getListeners](http://docs.google.com/java/awt/MenuItem.html#getListeners(java.lang.Class)) in class [MenuItem](http://docs.google.com/java/awt/MenuItem.html) **Parameters:**listenerType - the type of listeners requested; this parameter should specify an interface that descends from java.util.EventListener **Returns:**an array of all objects registered as *Foo*Listeners on this checkbox menuitem, or an empty array if no such listeners have been added **Throws:** [ClassCastException](http://docs.google.com/java/lang/ClassCastException.html) - if listenerType doesn't specify a class or interface that implements java.util.EventListener**Since:** 1.3 **See Also:**[getItemListeners()](http://docs.google.com/java/awt/CheckboxMenuItem.html#getItemListeners())

### processEvent

protected void **processEvent**([AWTEvent](http://docs.google.com/java/awt/AWTEvent.html) e)

Processes events on this check box menu item. If the event is an instance of ItemEvent, this method invokes the processItemEvent method. If the event is not an item event, it invokes processEvent on the superclass.

Check box menu items currently support only item events.

Note that if the event parameter is null the behavior is unspecified and may result in an exception.

**Overrides:**[processEvent](http://docs.google.com/java/awt/MenuItem.html#processEvent(java.awt.AWTEvent)) in class [MenuItem](http://docs.google.com/java/awt/MenuItem.html) **Parameters:**e - the event**Since:** JDK1.1 **See Also:**[ItemEvent](http://docs.google.com/java/awt/event/ItemEvent.html), [processItemEvent(java.awt.event.ItemEvent)](http://docs.google.com/java/awt/CheckboxMenuItem.html#processItemEvent(java.awt.event.ItemEvent))

### processItemEvent

protected void **processItemEvent**([ItemEvent](http://docs.google.com/java/awt/event/ItemEvent.html) e)

Processes item events occurring on this check box menu item by dispatching them to any registered ItemListener objects.

This method is not called unless item events are enabled for this menu item. Item events are enabled when one of the following occurs:

* An ItemListener object is registered via addItemListener.
* Item events are enabled via enableEvents.

Note that if the event parameter is null the behavior is unspecified and may result in an exception.

**Parameters:**e - the item event**Since:** JDK1.1 **See Also:**[ItemEvent](http://docs.google.com/java/awt/event/ItemEvent.html), [ItemListener](http://docs.google.com/java/awt/event/ItemListener.html), [addItemListener(java.awt.event.ItemListener)](http://docs.google.com/java/awt/CheckboxMenuItem.html#addItemListener(java.awt.event.ItemListener)), [MenuItem.enableEvents(long)](http://docs.google.com/java/awt/MenuItem.html#enableEvents(long))

### paramString

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

Returns a string representing the state of this CheckBoxMenuItem. This method is intended to be used only for debugging purposes, and the content and format of the returned string may vary between implementations. The returned string may be empty but may not be null.

**Overrides:**[paramString](http://docs.google.com/java/awt/MenuItem.html#paramString()) in class [MenuItem](http://docs.google.com/java/awt/MenuItem.html) **Returns:**the parameter string of this check box menu item

### getAccessibleContext

public [AccessibleContext](http://docs.google.com/javax/accessibility/AccessibleContext.html) **getAccessibleContext**()

Gets the AccessibleContext associated with this CheckboxMenuItem. For checkbox menu items, the AccessibleContext takes the form of an AccessibleAWTCheckboxMenuItem. A new AccessibleAWTCheckboxMenuItem is created if necessary.

**Specified by:**[getAccessibleContext](http://docs.google.com/javax/accessibility/Accessible.html#getAccessibleContext()) in interface [Accessible](http://docs.google.com/javax/accessibility/Accessible.html)**Overrides:**[getAccessibleContext](http://docs.google.com/java/awt/MenuItem.html#getAccessibleContext()) in class [MenuItem](http://docs.google.com/java/awt/MenuItem.html) **Returns:**an AccessibleAWTCheckboxMenuItem that serves as the AccessibleContext of this CheckboxMenuItem**Since:** 1.3

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/CheckboxMenuItem.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/java/awt/CheckboxGroup.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/CheckboxMenuItem.AccessibleAWTCheckboxMenuItem.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/CheckboxMenuItem.html)    [**NO FRAMES**](http://docs.google.com/CheckboxMenuItem.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | FIELD | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: FIELD | [CONSTR](#3rdcrjn) | [METHOD](#1ksv4uv) |

[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).