| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/MenuItem.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/MenuContainer.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/MenuItem.AccessibleAWTMenuItem.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/MenuItem.html)    [**NO FRAMES**](http://docs.google.com/MenuItem.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | FIELD | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#lnxbz9) |

## **java.awt**

Class MenuItem

[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**

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

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

All items in a menu must belong to the class MenuItem, or one of its subclasses.

The default MenuItem object embodies a simple labeled menu item.

This picture of a menu bar shows five menu items: 

The first two items are simple menu items, labeled "Basic" and "Simple". Following these two items is a separator, which is itself a menu item, created with the label "-". Next is an instance of CheckboxMenuItem labeled "Check". The final menu item is a submenu labeled "More Examples", and this submenu is an instance of Menu.

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

Note that the subclass Menu overrides this behavior and does not send any event to the frame until one of its subitems is selected.

**See Also:**[Serialized Form](http://docs.google.com/serialized-form.html#java.awt.MenuItem)

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

| **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** | |
| --- | --- |
| [**MenuItem**](http://docs.google.com/java/awt/MenuItem.html#MenuItem())()            Constructs a new MenuItem with an empty label and no keyboard shortcut. |
| [**MenuItem**](http://docs.google.com/java/awt/MenuItem.html#MenuItem(java.lang.String))([String](http://docs.google.com/java/lang/String.html) label)            Constructs a new MenuItem with the specified label and no keyboard shortcut. |
| [**MenuItem**](http://docs.google.com/java/awt/MenuItem.html#MenuItem(java.lang.String,%20java.awt.MenuShortcut))([String](http://docs.google.com/java/lang/String.html) label, [MenuShortcut](http://docs.google.com/java/awt/MenuShortcut.html) s)            Create a menu item with an associated keyboard shortcut. |

| **Method Summary** | |
| --- | --- |
| void | [**addActionListener**](http://docs.google.com/java/awt/MenuItem.html#addActionListener(java.awt.event.ActionListener))([ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) l)            Adds the specified action listener to receive action events from this menu item. |
| void | [**addNotify**](http://docs.google.com/java/awt/MenuItem.html#addNotify())()            Creates the menu item's peer. |
| void | [**deleteShortcut**](http://docs.google.com/java/awt/MenuItem.html#deleteShortcut())()            Delete any MenuShortcut object associated with this menu item. |
| void | [**disable**](http://docs.google.com/java/awt/MenuItem.html#disable())()  **Deprecated.** *As of JDK version 1.1, replaced by setEnabled(boolean).* |
| protected  void | [**disableEvents**](http://docs.google.com/java/awt/MenuItem.html#disableEvents(long))(long eventsToDisable)            Disables event delivery to this menu item for events defined by the specified event mask parameter. |
| void | [**enable**](http://docs.google.com/java/awt/MenuItem.html#enable())()  **Deprecated.** *As of JDK version 1.1, replaced by setEnabled(boolean).* |
| void | [**enable**](http://docs.google.com/java/awt/MenuItem.html#enable(boolean))(boolean b)  **Deprecated.** *As of JDK version 1.1, replaced by setEnabled(boolean).* |
| protected  void | [**enableEvents**](http://docs.google.com/java/awt/MenuItem.html#enableEvents(long))(long eventsToEnable)            Enables event delivery to this menu item for events to be defined by the specified event mask parameter |
| [AccessibleContext](http://docs.google.com/javax/accessibility/AccessibleContext.html) | [**getAccessibleContext**](http://docs.google.com/java/awt/MenuItem.html#getAccessibleContext())()            Gets the AccessibleContext associated with this MenuItem. |
| [String](http://docs.google.com/java/lang/String.html) | [**getActionCommand**](http://docs.google.com/java/awt/MenuItem.html#getActionCommand())()            Gets the command name of the action event that is fired by this menu item. |
| [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html)[] | [**getActionListeners**](http://docs.google.com/java/awt/MenuItem.html#getActionListeners())()            Returns an array of all the action listeners registered on this menu item. |
| [String](http://docs.google.com/java/lang/String.html) | [**getLabel**](http://docs.google.com/java/awt/MenuItem.html#getLabel())()            Gets the label for this menu item. |
| | <T extends [EventListener](http://docs.google.com/java/util/EventListener.html)>  T[] | | --- | | [**getListeners**](http://docs.google.com/java/awt/MenuItem.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 MenuItem. |
| [MenuShortcut](http://docs.google.com/java/awt/MenuShortcut.html) | [**getShortcut**](http://docs.google.com/java/awt/MenuItem.html#getShortcut())()            Get the MenuShortcut object associated with this menu item, |
| boolean | [**isEnabled**](http://docs.google.com/java/awt/MenuItem.html#isEnabled())()            Checks whether this menu item is enabled. |
| [String](http://docs.google.com/java/lang/String.html) | [**paramString**](http://docs.google.com/java/awt/MenuItem.html#paramString())()            Returns a string representing the state of this MenuItem. |
| protected  void | [**processActionEvent**](http://docs.google.com/java/awt/MenuItem.html#processActionEvent(java.awt.event.ActionEvent))([ActionEvent](http://docs.google.com/java/awt/event/ActionEvent.html) e)            Processes action events occurring on this menu item, by dispatching them to any registered ActionListener objects. |
| protected  void | [**processEvent**](http://docs.google.com/java/awt/MenuItem.html#processEvent(java.awt.AWTEvent))([AWTEvent](http://docs.google.com/java/awt/AWTEvent.html) e)            Processes events on this menu item. |
| void | [**removeActionListener**](http://docs.google.com/java/awt/MenuItem.html#removeActionListener(java.awt.event.ActionListener))([ActionListener](http://docs.google.com/java/awt/event/ActionListener.html) l)            Removes the specified action listener so it no longer receives action events from this menu item. |
| void | [**setActionCommand**](http://docs.google.com/java/awt/MenuItem.html#setActionCommand(java.lang.String))([String](http://docs.google.com/java/lang/String.html) command)            Sets the command name of the action event that is fired by this menu item. |
| void | [**setEnabled**](http://docs.google.com/java/awt/MenuItem.html#setEnabled(boolean))(boolean b)            Sets whether or not this menu item can be chosen. |
| void | [**setLabel**](http://docs.google.com/java/awt/MenuItem.html#setLabel(java.lang.String))([String](http://docs.google.com/java/lang/String.html) label)            Sets the label for this menu item to the specified label. |
| void | [**setShortcut**](http://docs.google.com/java/awt/MenuItem.html#setShortcut(java.awt.MenuShortcut))([MenuShortcut](http://docs.google.com/java/awt/MenuShortcut.html) s)            Set the MenuShortcut object associated with this menu item. |

| **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** |
| --- |

### MenuItem

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

Constructs a new MenuItem with an empty label and no keyboard shortcut.

**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())

### MenuItem

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

Constructs a new MenuItem with the specified label and no keyboard shortcut. Note that use of "-" in a label is reserved to indicate a separator between menu items. By default, all menu items except for separators are enabled.

**Parameters:**label - the label for this menu item. **Throws:** [HeadlessException](http://docs.google.com/java/awt/HeadlessException.html) - if GraphicsEnvironment.isHeadless() returns true.**Since:** JDK1.0 **See Also:**[GraphicsEnvironment.isHeadless()](http://docs.google.com/java/awt/GraphicsEnvironment.html#isHeadless())

### MenuItem

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

Create a menu item with an associated keyboard shortcut. Note that use of "-" in a label is reserved to indicate a separator between menu items. By default, all menu items except for separators are enabled.

**Parameters:**label - the label for this menu item.s - the instance of MenuShortcut associated with this menu item. **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 menu item's peer. The peer allows us to modify the appearance of the menu item without changing its functionality.

### getLabel

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

Gets the label for this menu item.

**Returns:**the label of this menu item, or null if this menu item has no label.**Since:** JDK1.0 **See Also:**[setLabel(java.lang.String)](http://docs.google.com/java/awt/MenuItem.html#setLabel(java.lang.String))

### setLabel

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

Sets the label for this menu item to the specified label.

**Parameters:**label - the new label, or null for no label.**Since:** JDK1.0 **See Also:**[getLabel()](http://docs.google.com/java/awt/MenuItem.html#getLabel())

### isEnabled

public boolean **isEnabled**()

Checks whether this menu item is enabled.

**Since:** JDK1.0 **See Also:**[setEnabled(boolean)](http://docs.google.com/java/awt/MenuItem.html#setEnabled(boolean))

### setEnabled

public void **setEnabled**(boolean b)

Sets whether or not this menu item can be chosen.

**Parameters:**b - if true, enables this menu item; if false, disables it.**Since:** JDK1.1 **See Also:**[isEnabled()](http://docs.google.com/java/awt/MenuItem.html#isEnabled())

### enable

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
public void **enable**()

**Deprecated.** *As of JDK version 1.1, replaced by setEnabled(boolean).*

### enable

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
public void **enable**(boolean b)

**Deprecated.** *As of JDK version 1.1, replaced by setEnabled(boolean).*

### disable

[@Deprecated](http://docs.google.com/java/lang/Deprecated.html)  
public void **disable**()

**Deprecated.** *As of JDK version 1.1, replaced by setEnabled(boolean).*

### getShortcut

public [MenuShortcut](http://docs.google.com/java/awt/MenuShortcut.html) **getShortcut**()

Get the MenuShortcut object associated with this menu item,

**Returns:**the menu shortcut associated with this menu item, or null if none has been specified.**Since:** JDK1.1 **See Also:**[setShortcut(java.awt.MenuShortcut)](http://docs.google.com/java/awt/MenuItem.html#setShortcut(java.awt.MenuShortcut))

### setShortcut

public void **setShortcut**([MenuShortcut](http://docs.google.com/java/awt/MenuShortcut.html) s)

Set the MenuShortcut object associated with this menu item. If a menu shortcut is already associated with this menu item, it is replaced.

**Parameters:**s - the menu shortcut to associate with this menu item.**Since:** JDK1.1 **See Also:**[getShortcut()](http://docs.google.com/java/awt/MenuItem.html#getShortcut())

### deleteShortcut

public void **deleteShortcut**()

Delete any MenuShortcut object associated with this menu item.

**Since:** JDK1.1

### enableEvents

protected final void **enableEvents**(long eventsToEnable)

Enables event delivery to this menu item for events to be defined by the specified event mask parameter

Since event types are automatically enabled when a listener for that type is added to the menu item, this method only needs to be invoked by subclasses of MenuItem which desire to have the specified event types delivered to processEvent regardless of whether a listener is registered.

**Parameters:**eventsToEnable - the event mask defining the event types**Since:** JDK1.1 **See Also:**[processEvent(java.awt.AWTEvent)](http://docs.google.com/java/awt/MenuItem.html#processEvent(java.awt.AWTEvent)), [disableEvents(long)](http://docs.google.com/java/awt/MenuItem.html#disableEvents(long)), [Component.enableEvents(long)](http://docs.google.com/java/awt/Component.html#enableEvents(long))

### disableEvents

protected final void **disableEvents**(long eventsToDisable)

Disables event delivery to this menu item for events defined by the specified event mask parameter.

**Parameters:**eventsToDisable - the event mask defining the event types**Since:** JDK1.1 **See Also:**[processEvent(java.awt.AWTEvent)](http://docs.google.com/java/awt/MenuItem.html#processEvent(java.awt.AWTEvent)), [enableEvents(long)](http://docs.google.com/java/awt/MenuItem.html#enableEvents(long)), [Component.disableEvents(long)](http://docs.google.com/java/awt/Component.html#disableEvents(long))

### setActionCommand

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

Sets the command name of the action event that is fired by this menu item.

By default, the action command is set to the label of the menu item.

**Parameters:**command - the action command to be set for this menu item.**Since:** JDK1.1 **See Also:**[getActionCommand()](http://docs.google.com/java/awt/MenuItem.html#getActionCommand())

### getActionCommand

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

Gets the command name of the action event that is fired by this menu item.

**Since:** JDK1.1 **See Also:**[setActionCommand(java.lang.String)](http://docs.google.com/java/awt/MenuItem.html#setActionCommand(java.lang.String))

### addActionListener

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

Adds the specified action listener to receive action events from this 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.

**Parameters:**l - the action listener.**Since:** JDK1.1 **See Also:**[removeActionListener(java.awt.event.ActionListener)](http://docs.google.com/java/awt/MenuItem.html#removeActionListener(java.awt.event.ActionListener)), [getActionListeners()](http://docs.google.com/java/awt/MenuItem.html#getActionListeners()), [ActionEvent](http://docs.google.com/java/awt/event/ActionEvent.html), [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html)

### removeActionListener

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

Removes the specified action listener so it no longer receives action events from this 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.

**Parameters:**l - the action listener.**Since:** JDK1.1 **See Also:**[addActionListener(java.awt.event.ActionListener)](http://docs.google.com/java/awt/MenuItem.html#addActionListener(java.awt.event.ActionListener)), [getActionListeners()](http://docs.google.com/java/awt/MenuItem.html#getActionListeners()), [ActionEvent](http://docs.google.com/java/awt/event/ActionEvent.html), [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html)

### getActionListeners

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

Returns an array of all the action listeners registered on this menu item.

**Returns:**all of this menu item's ActionListeners or an empty array if no action listeners are currently registered**Since:** 1.4 **See Also:**[addActionListener(java.awt.event.ActionListener)](http://docs.google.com/java/awt/MenuItem.html#addActionListener(java.awt.event.ActionListener)), [removeActionListener(java.awt.event.ActionListener)](http://docs.google.com/java/awt/MenuItem.html#removeActionListener(java.awt.event.ActionListener)), [ActionEvent](http://docs.google.com/java/awt/event/ActionEvent.html), [ActionListener](http://docs.google.com/java/awt/event/ActionListener.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 MenuItem. *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 MenuItem m for its action listeners with the following code:

ActionListener[] als = (ActionListener[])(m.getListeners(ActionListener.class));

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

**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 menu item, 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:**[getActionListeners()](http://docs.google.com/java/awt/MenuItem.html#getActionListeners())

### processEvent

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

Processes events on this menu item. If the event is an instance of ActionEvent, it invokes processActionEvent, another method defined by MenuItem.

Currently, menu items only support action 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/MenuComponent.html#processEvent(java.awt.AWTEvent)) in class [MenuComponent](http://docs.google.com/java/awt/MenuComponent.html) **Parameters:**e - the event**Since:** JDK1.1 **See Also:**[processActionEvent(java.awt.event.ActionEvent)](http://docs.google.com/java/awt/MenuItem.html#processActionEvent(java.awt.event.ActionEvent))

### processActionEvent

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

Processes action events occurring on this menu item, by dispatching them to any registered ActionListener objects. This method is not called unless action events are enabled for this component. Action events are enabled when one of the following occurs:

* An ActionListener object is registered via addActionListener.
* Action 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 action event**Since:** JDK1.1 **See Also:**[ActionEvent](http://docs.google.com/java/awt/event/ActionEvent.html), [ActionListener](http://docs.google.com/java/awt/event/ActionListener.html), [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 MenuItem. 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/MenuComponent.html#paramString()) in class [MenuComponent](http://docs.google.com/java/awt/MenuComponent.html) **Returns:**the parameter string of this menu item

### getAccessibleContext

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

Gets the AccessibleContext associated with this MenuItem. For menu items, the AccessibleContext takes the form of an AccessibleAWTMenuItem. A new AccessibleAWTMenuItem instance 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/MenuComponent.html#getAccessibleContext()) in class [MenuComponent](http://docs.google.com/java/awt/MenuComponent.html) **Returns:**an AccessibleAWTMenuItem that serves as the AccessibleContext of this MenuItem**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/MenuItem.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/MenuContainer.html)   [**NEXT CLASS**](http://docs.google.com/java/awt/MenuItem.AccessibleAWTMenuItem.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/MenuItem.html)    [**NO FRAMES**](http://docs.google.com/MenuItem.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: [NESTED](#3znysh7) | FIELD | [CONSTR](#tyjcwt) | [METHOD](#3dy6vkm) | DETAIL: FIELD | [CONSTR](#2s8eyo1) | [METHOD](#lnxbz9) |

[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).