| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ActionEvent.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   [**NEXT CLASS**](http://docs.google.com/java/awt/event/ActionListener.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/event/ActionEvent.html)    [**NO FRAMES**](http://docs.google.com/ActionEvent.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: [FIELD](#3rdcrjn) | [CONSTR](#3j2qqm3) | [METHOD](#1ci93xb) |

## **java.awt.event**

Class ActionEvent

[java.lang.Object](http://docs.google.com/java/lang/Object.html)  
 [java.util.EventObject](http://docs.google.com/java/util/EventObject.html)  
 [java.awt.AWTEvent](http://docs.google.com/java/awt/AWTEvent.html)  
 **java.awt.event.ActionEvent**

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

public class **ActionEvent**extends [AWTEvent](http://docs.google.com/java/awt/AWTEvent.html)

A semantic event which indicates that a component-defined action occurred. This high-level event is generated by a component (such as a Button) when the component-specific action occurs (such as being pressed). The event is passed to every every ActionListener object that registered to receive such events using the component's addActionListener method.

**Note:** To invoke an ActionEvent on a Button using the keyboard, use the Space bar.

The object that implements the ActionListener interface gets this ActionEvent when the event occurs. The listener is therefore spared the details of processing individual mouse movements and mouse clicks, and can instead process a "meaningful" (semantic) event like "button pressed".

**Since:** 1.1 **See Also:**[ActionListener](http://docs.google.com/java/awt/event/ActionListener.html), [Tutorial: Java 1.1 Event Model](http://java.sun.com/docs/books/tutorial/post1.0/ui/eventmodel.html), [Serialized Form](http://docs.google.com/serialized-form.html#java.awt.event.ActionEvent)

| **Field Summary** | |
| --- | --- |
| static int | [**ACTION\_FIRST**](http://docs.google.com/java/awt/event/ActionEvent.html#ACTION_FIRST)            The first number in the range of ids used for action events. |
| static int | [**ACTION\_LAST**](http://docs.google.com/java/awt/event/ActionEvent.html#ACTION_LAST)            The last number in the range of ids used for action events. |
| static int | [**ACTION\_PERFORMED**](http://docs.google.com/java/awt/event/ActionEvent.html#ACTION_PERFORMED)            This event id indicates that a meaningful action occured. |
| static int | [**ALT\_MASK**](http://docs.google.com/java/awt/event/ActionEvent.html#ALT_MASK)            The alt modifier. |
| static int | [**CTRL\_MASK**](http://docs.google.com/java/awt/event/ActionEvent.html#CTRL_MASK)            The control modifier. |
| static int | [**META\_MASK**](http://docs.google.com/java/awt/event/ActionEvent.html#META_MASK)            The meta modifier. |
| static int | [**SHIFT\_MASK**](http://docs.google.com/java/awt/event/ActionEvent.html#SHIFT_MASK)            The shift modifier. |

| **Fields inherited from class java.awt.**[**AWTEvent**](http://docs.google.com/java/awt/AWTEvent.html) |
| --- |
| [ACTION\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#ACTION_EVENT_MASK), [ADJUSTMENT\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#ADJUSTMENT_EVENT_MASK), [COMPONENT\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#COMPONENT_EVENT_MASK), [consumed](http://docs.google.com/java/awt/AWTEvent.html#consumed), [CONTAINER\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#CONTAINER_EVENT_MASK), [FOCUS\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#FOCUS_EVENT_MASK), [HIERARCHY\_BOUNDS\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#HIERARCHY_BOUNDS_EVENT_MASK), [HIERARCHY\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#HIERARCHY_EVENT_MASK), [id](http://docs.google.com/java/awt/AWTEvent.html#id), [INPUT\_METHOD\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#INPUT_METHOD_EVENT_MASK), [INVOCATION\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#INVOCATION_EVENT_MASK), [ITEM\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#ITEM_EVENT_MASK), [KEY\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#KEY_EVENT_MASK), [MOUSE\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#MOUSE_EVENT_MASK), [MOUSE\_MOTION\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#MOUSE_MOTION_EVENT_MASK), [MOUSE\_WHEEL\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#MOUSE_WHEEL_EVENT_MASK), [PAINT\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#PAINT_EVENT_MASK), [RESERVED\_ID\_MAX](http://docs.google.com/java/awt/AWTEvent.html#RESERVED_ID_MAX), [TEXT\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#TEXT_EVENT_MASK), [WINDOW\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#WINDOW_EVENT_MASK), [WINDOW\_FOCUS\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#WINDOW_FOCUS_EVENT_MASK), [WINDOW\_STATE\_EVENT\_MASK](http://docs.google.com/java/awt/AWTEvent.html#WINDOW_STATE_EVENT_MASK) |

| **Fields inherited from class java.util.**[**EventObject**](http://docs.google.com/java/util/EventObject.html) |
| --- |
| [source](http://docs.google.com/java/util/EventObject.html#source) |

| **Constructor Summary** | |
| --- | --- |
| [**ActionEvent**](http://docs.google.com/java/awt/event/ActionEvent.html#ActionEvent(java.lang.Object,%20int,%20java.lang.String))([Object](http://docs.google.com/java/lang/Object.html) source, int id, [String](http://docs.google.com/java/lang/String.html) command)            Constructs an ActionEvent object. |
| [**ActionEvent**](http://docs.google.com/java/awt/event/ActionEvent.html#ActionEvent(java.lang.Object,%20int,%20java.lang.String,%20int))([Object](http://docs.google.com/java/lang/Object.html) source, int id, [String](http://docs.google.com/java/lang/String.html) command, int modifiers)            Constructs an ActionEvent object with modifier keys. |
| [**ActionEvent**](http://docs.google.com/java/awt/event/ActionEvent.html#ActionEvent(java.lang.Object,%20int,%20java.lang.String,%20long,%20int))([Object](http://docs.google.com/java/lang/Object.html) source, int id, [String](http://docs.google.com/java/lang/String.html) command, long when, int modifiers)            Constructs an ActionEvent object with the specified modifier keys and timestamp. |

| **Method Summary** | |
| --- | --- |
| [String](http://docs.google.com/java/lang/String.html) | [**getActionCommand**](http://docs.google.com/java/awt/event/ActionEvent.html#getActionCommand())()            Returns the command string associated with this action. |
| int | [**getModifiers**](http://docs.google.com/java/awt/event/ActionEvent.html#getModifiers())()            Returns the modifier keys held down during this action event. |
| long | [**getWhen**](http://docs.google.com/java/awt/event/ActionEvent.html#getWhen())()            Returns the timestamp of when this event occurred. |
| [String](http://docs.google.com/java/lang/String.html) | [**paramString**](http://docs.google.com/java/awt/event/ActionEvent.html#paramString())()            Returns a parameter string identifying this action event. |

| **Methods inherited from class java.awt.**[**AWTEvent**](http://docs.google.com/java/awt/AWTEvent.html) |
| --- |
| [consume](http://docs.google.com/java/awt/AWTEvent.html#consume()), [getID](http://docs.google.com/java/awt/AWTEvent.html#getID()), [isConsumed](http://docs.google.com/java/awt/AWTEvent.html#isConsumed()), [setSource](http://docs.google.com/java/awt/AWTEvent.html#setSource(java.lang.Object)), [toString](http://docs.google.com/java/awt/AWTEvent.html#toString()) |

| **Methods inherited from class java.util.**[**EventObject**](http://docs.google.com/java/util/EventObject.html) |
| --- |
| [getSource](http://docs.google.com/java/util/EventObject.html#getSource()) |

| **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)) |

| **Field Detail** |
| --- |

### SHIFT\_MASK

public static final int **SHIFT\_MASK**

The shift modifier. An indicator that the shift key was held down during the event.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.event.ActionEvent.SHIFT_MASK)

### CTRL\_MASK

public static final int **CTRL\_MASK**

The control modifier. An indicator that the control key was held down during the event.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.event.ActionEvent.CTRL_MASK)

### META\_MASK

public static final int **META\_MASK**

The meta modifier. An indicator that the meta key was held down during the event.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.event.ActionEvent.META_MASK)

### ALT\_MASK

public static final int **ALT\_MASK**

The alt modifier. An indicator that the alt key was held down during the event.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.event.ActionEvent.ALT_MASK)

### ACTION\_FIRST

public static final int **ACTION\_FIRST**

The first number in the range of ids used for action events.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.event.ActionEvent.ACTION_FIRST)

### ACTION\_LAST

public static final int **ACTION\_LAST**

The last number in the range of ids used for action events.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.event.ActionEvent.ACTION_LAST)

### ACTION\_PERFORMED

public static final int **ACTION\_PERFORMED**

This event id indicates that a meaningful action occured.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#java.awt.event.ActionEvent.ACTION_PERFORMED)

| **Constructor Detail** |
| --- |

### ActionEvent

public **ActionEvent**([Object](http://docs.google.com/java/lang/Object.html) source,  
 int id,  
 [String](http://docs.google.com/java/lang/String.html) command)

Constructs an ActionEvent object.

Note that passing in an invalid id results in unspecified behavior. This method throws an IllegalArgumentException if source is null. A null command string is legal, but not recommended.

**Parameters:**source - the object that originated the eventid - an integer that identifies the eventcommand - a string that may specify a command (possibly one of several) associated with the event **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if source is null

### ActionEvent

public **ActionEvent**([Object](http://docs.google.com/java/lang/Object.html) source,  
 int id,  
 [String](http://docs.google.com/java/lang/String.html) command,  
 int modifiers)

Constructs an ActionEvent object with modifier keys.

Note that passing in an invalid id results in unspecified behavior. This method throws an IllegalArgumentException if source is null. A null command string is legal, but not recommended.

**Parameters:**source - the object that originated the eventid - an integer that identifies the eventcommand - a string that may specify a command (possibly one of several) associated with the eventmodifiers - the modifier keys held down during this action **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if source is null

### ActionEvent

public **ActionEvent**([Object](http://docs.google.com/java/lang/Object.html) source,  
 int id,  
 [String](http://docs.google.com/java/lang/String.html) command,  
 long when,  
 int modifiers)

Constructs an ActionEvent object with the specified modifier keys and timestamp.

Note that passing in an invalid id results in unspecified behavior. This method throws an IllegalArgumentException if source is null. A null command string is legal, but not recommended.

**Parameters:**source - the object that originated the eventid - an integer that identifies the eventcommand - a string that may specify a command (possibly one of several) associated with the eventwhen - the time the event occurredmodifiers - the modifier keys held down during this action **Throws:** [IllegalArgumentException](http://docs.google.com/java/lang/IllegalArgumentException.html) - if source is null**Since:** 1.4

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

### getActionCommand

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

Returns the command string associated with this action. This string allows a "modal" component to specify one of several commands, depending on its state. For example, a single button might toggle between "show details" and "hide details". The source object and the event would be the same in each case, but the command string would identify the intended action.

Note that if a null command string was passed to the constructor for this ActionEvent, this this method returns null.

**Returns:**the string identifying the command for this event

### getWhen

public long **getWhen**()

Returns the timestamp of when this event occurred. Because an ActionEvent is a high-level, semantic event, the timestamp is typically the same as an underlying InputEvent.

**Returns:**this event's timestamp**Since:** 1.4

### getModifiers

public int **getModifiers**()

Returns the modifier keys held down during this action event.

**Returns:**the bitwise-or of the modifier constants

### paramString

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

Returns a parameter string identifying this action event. This method is useful for event-logging and for debugging.

**Overrides:**[paramString](http://docs.google.com/java/awt/AWTEvent.html#paramString()) in class [AWTEvent](http://docs.google.com/java/awt/AWTEvent.html) **Returns:**a string identifying the event and its associated command

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/ActionEvent.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   [**NEXT CLASS**](http://docs.google.com/java/awt/event/ActionListener.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/event/ActionEvent.html)    [**NO FRAMES**](http://docs.google.com/ActionEvent.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#3dy6vkm) | [METHOD](#1t3h5sf) | DETAIL: [FIELD](#3rdcrjn) | [CONSTR](#3j2qqm3) | [METHOD](#1ci93xb) |

[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).