Class java.awt.Event

```
java.lang.Object
   |
   +----java.awt.Event
```

---

public class **Event**
extends Object

Event is a platform−independent class that encapsulates events from the local Graphical User Interface(GUI) platform.

Version:
1.27 10/02/95
Author:
Sami Shaio

Variable Index

o **ACTION_EVENT**
An action event.
o **ALT_MASK**
The alt modifier constant.
o **CTRL_MASK**
The control modifier constant.
o **DOWN**
The down arrow key.
o **END**
The end key.
o **ESC**
The ESC key
o **F1**
The F1 function key
o **F10**
The F10 function key
o **F11**
The F11 function key
o **F12**

The F12 function key

- **F2**
  The F2 function key
- **F3**
  The F3 function key
- **F4**
  The F4 function key
- **F5**
  The F5 function key
- **F6**
  The F6 function key
- **F7**
  The F7 function key
- **F8**
  The F8 function key
- **F9**
  The F9 function key
- **GOT_FOCUS**
  A component gained the focus.
- **HOME**
  The home key.
- **KEY_ACTION**
  The key action keyboard event.
- **KEY_ACTION_RELEASE**
  The key action keyboard event.
- **KEY_PRESS**
  The key press keyboard event.
- **KEY_RELEASE**
  The key release keyboard event.
- **LEFT**
  The left arrow key.
- **LIST_DESELECT**
- **LIST_SELECT**
- **LOAD_FILE**
  A file loading event.
- **LOST_FOCUS**
  A component lost the focus.
- **META_MASK**
  The meta modifier constant.
- **MOUSE_DOWN**
  The mouse down event.
- **MOUSE_DRAG**
  The mouse drag event.
- **MOUSE_ENTER**
  The mouse enter event.
- **MOUSE_EXIT**
  The mouse exit event.
- **MOUSE_MOVE**
  The mouse move event.

- **MOUSE_UP**
  The mouse up event.
- **PGDN**
  The page down key.
- **PGUP**
  The page up key.
- **RIGHT**
  The right arrow key.
- **SAVE_FILE**
  A file saving event.
- **SCROLL_ABSOLUTE**
  The absolute scroll event.
- **SCROLL_LINE_DOWN**
  The line down scroll event.
- **SCROLL_LINE_UP**
  The line up scroll event.
- **SCROLL_PAGE_DOWN**
  The page down scroll event.
- **SCROLL_PAGE_UP**
  The page up scroll event.
- **SHIFT_MASK**
  The shift modifier constant.
- **UP**
  The up arrow key.
- **WINDOW_DEICONIFY**
  The de−iconify window event.
- **WINDOW_DESTROY**
  The destroy window event.
- **WINDOW_EXPOSE**
  The expose window event.
- **WINDOW_ICONIFY**
  The iconify window event.
- **WINDOW_MOVED**
  The move window event.
- **arg**
  An arbitraty argument.
- **evt**
  The next event.
- **id**
  The type of this event.
- **key**
  The key that was pressed in a keyboard event.
- **modifiers**
  The state of the modifier keys.
- **target**
  The target component.
- **when**
  The time stamp.
- **x**

The x coordinate of the event.

- **y**
  The y coordinate of the event.

Constructor Index

- **Event**(Object, long, int, int, int, int, int, Object)
  Constructs an event with the specified target component, time stamp, event type, x and y coordinates, keyboard key, state of the modifier keys and argument.
- **Event**(Object, long, int, int, int, int, int)
  Constructs an event with the specified target component, time stamp, event type, x and y coordinates, keyboard key, state of the modifier keys and an argument set to null.
- **Event**(Object, int, Object)
  Constructs an event with the specified target component, event type, and argument.

Method Index

- **controlDown**()
  Checks if the control key is down.
- **metaDown**()
  Checks if the meta key is down.
- **paramString**()
  Returns the parameter String of this Event.
- **shiftDown**()
  Checks if the shift key is down.
- **toString**()
  Returns the String representation of this Event's values.
- **translate**(int, int)
  Translates an event relative to the given component.

Variables

- **SHIFT_MASK**

```
public final static int SHIFT_MASK
```

The shift modifier constant.

- **CTRL_MASK**

```
public final static int CTRL_MASK
```

The control modifier constant.

- **META_MASK**

```
public final static int META_MASK
```

The meta modifier constant.

o **ALT_MASK**

```
public final static int ALT_MASK
```

The alt modifier constant.

o **HOME**

```
public final static int HOME
```

The home key.

o **END**

```
public final static int END
```

The end key.

o **PGUP**

```
public final static int PGUP
```

The page up key.

o **PGDN**

```
public final static int PGDN
```

The page down key.

o **UP**

```
public final static int UP
```

The up arrow key.

o **DOWN**

```
public final static int DOWN
```

The down arrow key.

o **LEFT**

```
public final static int LEFT
```

The left arrow key.

o **RIGHT**

```
public final static int RIGHT
```

The right arrow key.

o **F1**

```
public final static int F1
```

The F1 function key

o **F2**

```
public final static int F2
```

The F2 function key

o **F3**

```
public final static int F3
```

The F3 function key

o **F4**

```
public final static int F4
```

The F4 function key

o **F5**

```
public final static int F5
```

The F5 function key

o **F6**

```
public final static int F6
```

The F6 function key

o **F7**

```
public final static int F7
```

The F7 function key

o **F8**

```
public final static int F8
```

The F8 function key

- **F9**

```
public final static int F9
```

The F9 function key

- **F10**

```
public final static int F10
```

The F10 function key

- **F11**

```
public final static int F11
```

The F11 function key

- **F12**

```
public final static int F12
```

The F12 function key

- **ESC**

```
public final static int ESC
```

The ESC key

- **WINDOW_DESTROY**

```
public final static int WINDOW_DESTROY
```

The destroy window event.

- **WINDOW_EXPOSE**

```
public final static int WINDOW_EXPOSE
```

The expose window event.

- **WINDOW_ICONIFY**

```
public final static int WINDOW_ICONIFY
```

The iconify window event.

- **WINDOW_DEICONIFY**

```
public final static int WINDOW_DEICONIFY
```

The de−iconify window event.

- **WINDOW_MOVED**

```
public final static int WINDOW_MOVED
```

The move window event.

- **KEY_PRESS**

```
public final static int KEY_PRESS
```

The key press keyboard event.

- **KEY_RELEASE**

```
public final static int KEY_RELEASE
```

The key release keyboard event.

- **KEY_ACTION**

```
public final static int KEY_ACTION
```

The key action keyboard event.

- **KEY_ACTION_RELEASE**

```
public final static int KEY_ACTION_RELEASE
```

The key action keyboard event.

- **MOUSE_DOWN**

```
public final static int MOUSE_DOWN
```

The mouse down event.

- **MOUSE_UP**

```
public final static int MOUSE_UP
```

The mouse up event.

- **MOUSE_MOVE**

```
public final static int MOUSE_MOVE
```

The mouse move event.

- **MOUSE_ENTER**

```
public final static int MOUSE_ENTER
```

The mouse enter event.

- **MOUSE_EXIT**

```
public final static int MOUSE_EXIT
```

The mouse exit event.

- **MOUSE_DRAG**

```
public final static int MOUSE_DRAG
```

The mouse drag event.

- **SCROLL_LINE_UP**

```
public final static int SCROLL_LINE_UP
```

The line up scroll event.

- **SCROLL_LINE_DOWN**

```
public final static int SCROLL_LINE_DOWN
```

The line down scroll event.

- **SCROLL_PAGE_UP**

```
public final static int SCROLL_PAGE_UP
```

The page up scroll event.

- **SCROLL_PAGE_DOWN**

```
public final static int SCROLL_PAGE_DOWN
```

The page down scroll event.

o **SCROLL_ABSOLUTE**

```
public final static int SCROLL_ABSOLUTE
```

The absolute scroll event.

o **LIST_SELECT**

```
public final static int LIST_SELECT
```

o **LIST_DESELECT**

```
public final static int LIST_DESELECT
```

o **ACTION_EVENT**

```
public final static int ACTION_EVENT
```

An action event.

o **LOAD_FILE**

```
public final static int LOAD_FILE
```

A file loading event.

o **SAVE_FILE**

```
public final static int SAVE_FILE
```

A file saving event.

o **GOT_FOCUS**

```
public final static int GOT_FOCUS
```

A component gained the focus.

o **LOST_FOCUS**

```
public final static int LOST_FOCUS
```

A component lost the focus.

o **target**

```
public Object target
```

The target component.

o **when**

```
public long when
```

The time stamp.

o **id**

```
public int id
```

The type of this event.

o **x**

```
public int x
```

The x coordinate of the event.

o **y**

```
public int y
```

The y coordinate of the event.

o **key**

```
public int key
```

The key that was pressed in a keyboard event.

o **modifiers**

```
public int modifiers
```

The state of the modifier keys.

o **arg**

```
public Object arg
```

An arbitraty argument.

o **evt**

```
public Event evt
```

The next event. Used when putting events into a linked list.

Constructors

o **Event**

```
public Event(Object target,
             long when,
             int id,
             int x,
             int y,
             int key,
             int modifiers,
             Object arg)
```

Constructs an event with the specified target component, time stamp, event type, x and y coordinates, keyboard key, state of the modifier keys and argument.

Parameters:

target – the target component
when – the time stamp
id – the event type
x – the x coordinate
y – the y coordinate
key – the key pressed in a keyboard event
modifiers – the state of the modifier keys
arg – the specified argument

o **Event**

```
public Event(Object target,
             long when,
             int id,
             int x,
             int y,
             int key,
             int modifiers)
```

Constructs an event with the specified target component, time stamp, event type, x and y coordinates, keyboard key, state of the modifier keys and an argument set to null.

Parameters:

target – the target component
when – the time stamp
id – the event type
x – the x coordinate
y – the y coordinate
key – the key pressed in a keyboard event
modifiers – the state of the modifier keys

o **Event**

```
public Event(Object target,
             int id,
             Object arg)
```

Constructs an event with the specified target component, event type, and argument.
Parameters:
target – the target component
id – the event type
arg – the specified argument

Methods

o **translate**

```
public void translate(int x,
                      int y)
```

Translates an event relative to the given component. This involves at a minimum translating the coordinates so they make sense within the given component. It may also involve translating a region in the case of an expose event.
Parameters:
x – the x coordinate
y – the y coordinate

o **shiftDown**

```
public boolean shiftDown()
```

Checks if the shift key is down.
See Also:
modifiers, controlDown, metaDown

o **controlDown**

```
public boolean controlDown()
```

Checks if the control key is down.
See Also:
modifiers, shiftDown, metaDown

o **metaDown**

```
public boolean metaDown()
```

Checks if the meta key is down.
See Also:
modifiers, shiftDown, controlDown

o **paramString**

```
protected String paramString()
```

Returns the parameter String of this Event.

o **toString**

```
public String toString()
```

Returns the String representation of this Event's values.
Overrides:
toString in class Object