| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/awt/event/MouseEvent.html) | **Use** | [**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   NEXT | [**FRAMES**](http://docs.google.com/index.html?java/awt/event//class-useMouseEvent.html)    [**NO FRAMES**](http://docs.google.com/MouseEvent.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

**Uses of Class**

**java.awt.event.MouseEvent**

| Packages that use [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| [**java.awt**](#3znysh7) | Contains all of the classes for creating user interfaces and for painting graphics and images. |
| [**java.awt.dnd**](#2et92p0) | Drag and Drop is a direct manipulation gesture found in many Graphical User Interface systems that provides a mechanism to transfer information between two entities logically associated with presentation elements in the GUI. |
| [**java.awt.event**](#tyjcwt) | Provides interfaces and classes for dealing with different types of events fired by AWT components. |
| [**javax.swing**](#3dy6vkm) | Provides a set of "lightweight" (all-Java language) components that, to the maximum degree possible, work the same on all platforms. |
| [**javax.swing.event**](#1t3h5sf) | Provides for events fired by Swing components. |
| [**javax.swing.plaf**](#4d34og8) | Provides one interface and many abstract classes that Swing uses to provide its pluggable look-and-feel capabilities. |
| [**javax.swing.plaf.basic**](#2s8eyo1) | Provides user interface objects built according to the Basic look and feel. |
| [**javax.swing.plaf.metal**](#17dp8vu) | Provides user interface objects built according to the Java look and feel (once codenamed *Metal*), which is the default look and feel. |
| [**javax.swing.plaf.multi**](#3rdcrjn) | Provides user interface objects that combine two or more look and feels. |
| [**javax.swing.table**](#26in1rg) | Provides classes and interfaces for dealing with javax.swing.JTable. |
| [**javax.swing.text**](#lnxbz9) | Provides classes and interfaces that deal with editable and noneditable text components. |
| [**javax.swing.text.html**](#35nkun2) | Provides the class HTMLEditorKit and supporting classes for creating HTML text editors. |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [java.awt](http://docs.google.com/java/awt/package-summary.html) | |
| --- | --- |

| Methods in [java.awt](http://docs.google.com/java/awt/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| void | **AWTEventMulticaster.**[**mouseClicked**](http://docs.google.com/java/awt/AWTEventMulticaster.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Handles the mouseClicked event by invoking the mouseClicked methods on listener-a and listener-b. |
| void | **AWTEventMulticaster.**[**mouseDragged**](http://docs.google.com/java/awt/AWTEventMulticaster.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Handles the mouseDragged event by invoking the mouseDragged methods on listener-a and listener-b. |
| void | **AWTEventMulticaster.**[**mouseEntered**](http://docs.google.com/java/awt/AWTEventMulticaster.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Handles the mouseEntered event by invoking the mouseEntered methods on listener-a and listener-b. |
| void | **AWTEventMulticaster.**[**mouseExited**](http://docs.google.com/java/awt/AWTEventMulticaster.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Handles the mouseExited event by invoking the mouseExited methods on listener-a and listener-b. |
| void | **AWTEventMulticaster.**[**mouseMoved**](http://docs.google.com/java/awt/AWTEventMulticaster.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Handles the mouseMoved event by invoking the mouseMoved methods on listener-a and listener-b. |
| void | **AWTEventMulticaster.**[**mousePressed**](http://docs.google.com/java/awt/AWTEventMulticaster.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Handles the mousePressed event by invoking the mousePressed methods on listener-a and listener-b. |
| void | **AWTEventMulticaster.**[**mouseReleased**](http://docs.google.com/java/awt/AWTEventMulticaster.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Handles the mouseReleased event by invoking the mouseReleased methods on listener-a and listener-b. |
| protected  void | **Component.**[**processMouseEvent**](http://docs.google.com/java/awt/Component.html#processMouseEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Processes mouse events occurring on this component by dispatching them to any registered MouseListener objects. |
| protected  void | **Component.**[**processMouseMotionEvent**](http://docs.google.com/java/awt/Component.html#processMouseMotionEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Processes mouse motion events occurring on this component by dispatching them to any registered MouseMotionListener objects. |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [java.awt.dnd](http://docs.google.com/java/awt/dnd/package-summary.html) | |
| --- | --- |

| Methods in [java.awt.dnd](http://docs.google.com/java/awt/dnd/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| void | **MouseDragGestureRecognizer.**[**mouseClicked**](http://docs.google.com/java/awt/dnd/MouseDragGestureRecognizer.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse has been clicked on a component. |
| void | **MouseDragGestureRecognizer.**[**mouseDragged**](http://docs.google.com/java/awt/dnd/MouseDragGestureRecognizer.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button is pressed on a component. |
| void | **MouseDragGestureRecognizer.**[**mouseEntered**](http://docs.google.com/java/awt/dnd/MouseDragGestureRecognizer.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse enters a component. |
| void | **MouseDragGestureRecognizer.**[**mouseExited**](http://docs.google.com/java/awt/dnd/MouseDragGestureRecognizer.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse exits a component. |
| void | **MouseDragGestureRecognizer.**[**mouseMoved**](http://docs.google.com/java/awt/dnd/MouseDragGestureRecognizer.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse button has been moved on a component (with no buttons no down). |
| void | **MouseDragGestureRecognizer.**[**mousePressed**](http://docs.google.com/java/awt/dnd/MouseDragGestureRecognizer.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button has been pressed on a Component. |
| void | **MouseDragGestureRecognizer.**[**mouseReleased**](http://docs.google.com/java/awt/dnd/MouseDragGestureRecognizer.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button has been released on a component. |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [java.awt.event](http://docs.google.com/java/awt/event/package-summary.html) | |
| --- | --- |

| Subclasses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [java.awt.event](http://docs.google.com/java/awt/event/package-summary.html) | |
| --- | --- |
| class | [**MouseWheelEvent**](http://docs.google.com/java/awt/event/MouseWheelEvent.html)            An event which indicates that the mouse wheel was rotated in a component. |

| Methods in [java.awt.event](http://docs.google.com/java/awt/event/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| void | **MouseAdapter.**[**mouseClicked**](http://docs.google.com/java/awt/event/MouseAdapter.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse button has been clicked (pressed and released) on a component. |
| void | **MouseListener.**[**mouseClicked**](http://docs.google.com/java/awt/event/MouseListener.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse button has been clicked (pressed and released) on a component. |
| void | **MouseMotionAdapter.**[**mouseDragged**](http://docs.google.com/java/awt/event/MouseMotionAdapter.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button is pressed on a component and then dragged. |
| void | **MouseAdapter.**[**mouseDragged**](http://docs.google.com/java/awt/event/MouseAdapter.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button is pressed on a component and then dragged. |
| void | **MouseMotionListener.**[**mouseDragged**](http://docs.google.com/java/awt/event/MouseMotionListener.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button is pressed on a component and then dragged. |
| void | **MouseAdapter.**[**mouseEntered**](http://docs.google.com/java/awt/event/MouseAdapter.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse enters a component. |
| void | **MouseListener.**[**mouseEntered**](http://docs.google.com/java/awt/event/MouseListener.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse enters a component. |
| void | **MouseAdapter.**[**mouseExited**](http://docs.google.com/java/awt/event/MouseAdapter.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse exits a component. |
| void | **MouseListener.**[**mouseExited**](http://docs.google.com/java/awt/event/MouseListener.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse exits a component. |
| void | **MouseMotionAdapter.**[**mouseMoved**](http://docs.google.com/java/awt/event/MouseMotionAdapter.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse button has been moved on a component (with no buttons no down). |
| void | **MouseAdapter.**[**mouseMoved**](http://docs.google.com/java/awt/event/MouseAdapter.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse cursor has been moved onto a component but no buttons have been pushed. |
| void | **MouseMotionListener.**[**mouseMoved**](http://docs.google.com/java/awt/event/MouseMotionListener.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse cursor has been moved onto a component but no buttons have been pushed. |
| void | **MouseAdapter.**[**mousePressed**](http://docs.google.com/java/awt/event/MouseAdapter.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button has been pressed on a component. |
| void | **MouseListener.**[**mousePressed**](http://docs.google.com/java/awt/event/MouseListener.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button has been pressed on a component. |
| void | **MouseAdapter.**[**mouseReleased**](http://docs.google.com/java/awt/event/MouseAdapter.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button has been released on a component. |
| void | **MouseListener.**[**mouseReleased**](http://docs.google.com/java/awt/event/MouseListener.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button has been released on a component. |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) | |
| --- | --- |

| Methods in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) that return [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| static [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | **SwingUtilities.**[**convertMouseEvent**](http://docs.google.com/javax/swing/SwingUtilities.html#convertMouseEvent(java.awt.Component,%20java.awt.event.MouseEvent,%20java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) source, [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) sourceEvent, [Component](http://docs.google.com/java/awt/Component.html) destination)            Returns a MouseEvent similar to sourceEvent except that its x and y members have been converted to destination's coordinate system. |

| Methods in [javax.swing](http://docs.google.com/javax/swing/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| static [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | **SwingUtilities.**[**convertMouseEvent**](http://docs.google.com/javax/swing/SwingUtilities.html#convertMouseEvent(java.awt.Component,%20java.awt.event.MouseEvent,%20java.awt.Component))([Component](http://docs.google.com/java/awt/Component.html) source, [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) sourceEvent, [Component](http://docs.google.com/java/awt/Component.html) destination)            Returns a MouseEvent similar to sourceEvent except that its x and y members have been converted to destination's coordinate system. |
| [Point](http://docs.google.com/java/awt/Point.html) | **JComponent.**[**getPopupLocation**](http://docs.google.com/javax/swing/JComponent.html#getPopupLocation(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Returns the preferred location to display the popup menu in this component's coordinate system. |
| [Point](http://docs.google.com/java/awt/Point.html) | **JComponent.**[**getToolTipLocation**](http://docs.google.com/javax/swing/JComponent.html#getToolTipLocation(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Returns the tooltip location in this component's coordinate system. |
| [String](http://docs.google.com/java/lang/String.html) | **JTabbedPane.**[**getToolTipText**](http://docs.google.com/javax/swing/JTabbedPane.html#getToolTipText(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Returns the tooltip text for the component determined by the mouse event location. |
| [String](http://docs.google.com/java/lang/String.html) | **JTree.**[**getToolTipText**](http://docs.google.com/javax/swing/JTree.html#getToolTipText(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Overrides JComponent's getToolTipText method in order to allow renderer's tips to be used if it has text set. |
| [String](http://docs.google.com/java/lang/String.html) | **JTable.**[**getToolTipText**](http://docs.google.com/javax/swing/JTable.html#getToolTipText(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Overrides JComponent's getToolTipText method in order to allow the renderer's tips to be used if it has text set. |
| [String](http://docs.google.com/java/lang/String.html) | **JList.**[**getToolTipText**](http://docs.google.com/javax/swing/JList.html#getToolTipText(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Returns the tooltip text to be used for the given event. |
| [String](http://docs.google.com/java/lang/String.html) | **JComponent.**[**getToolTipText**](http://docs.google.com/javax/swing/JComponent.html#getToolTipText(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Returns the string to be used as the tooltip for *event*. |
| static boolean | **SwingUtilities.**[**isLeftMouseButton**](http://docs.google.com/javax/swing/SwingUtilities.html#isLeftMouseButton(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) anEvent)            Returns true if the mouse event specifies the left mouse button. |
| static boolean | **SwingUtilities.**[**isMiddleMouseButton**](http://docs.google.com/javax/swing/SwingUtilities.html#isMiddleMouseButton(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) anEvent)            Returns true if the mouse event specifies the middle mouse button. |
| boolean | **JPopupMenu.**[**isPopupTrigger**](http://docs.google.com/javax/swing/JPopupMenu.html#isPopupTrigger(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Returns true if the MouseEvent is considered a popup trigger by the JPopupMenu's currently installed UI. |
| static boolean | **SwingUtilities.**[**isRightMouseButton**](http://docs.google.com/javax/swing/SwingUtilities.html#isRightMouseButton(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) anEvent)            Returns true if the mouse event specifies the right mouse button. |
| void | **ToolTipManager.**[**mouseDragged**](http://docs.google.com/javax/swing/ToolTipManager.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Called when the mouse is pressed and dragged. |
| void | **ToolTipManager.**[**mouseEntered**](http://docs.google.com/javax/swing/ToolTipManager.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Called when the mouse enters the region of a component. |
| void | **ToolTipManager.**[**mouseExited**](http://docs.google.com/javax/swing/ToolTipManager.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Called when the mouse exits the region of a component. |
| void | **ToolTipManager.**[**mouseMoved**](http://docs.google.com/javax/swing/ToolTipManager.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Called when the mouse is moved. |
| void | **ToolTipManager.**[**mousePressed**](http://docs.google.com/javax/swing/ToolTipManager.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Called when the mouse is pressed. |
| protected  void | **JComponent.**[**processMouseEvent**](http://docs.google.com/javax/swing/JComponent.html#processMouseEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Processes mouse events occurring on this component by dispatching them to any registered MouseListener objects, refer to [Component.processMouseEvent(MouseEvent)](http://docs.google.com/java/awt/Component.html#processMouseEvent(java.awt.event.MouseEvent)) for a complete description of this method. |
| void | **MenuSelectionManager.**[**processMouseEvent**](http://docs.google.com/javax/swing/MenuSelectionManager.html#processMouseEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            When a MenuElement receives an event from a MouseListener, it should never process the event directly. |
| void | **JMenuItem.**[**processMouseEvent**](http://docs.google.com/javax/swing/JMenuItem.html#processMouseEvent(java.awt.event.MouseEvent,%20javax.swing.MenuElement%5B%5D,%20javax.swing.MenuSelectionManager))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e, [MenuElement](http://docs.google.com/javax/swing/MenuElement.html)[] path, [MenuSelectionManager](http://docs.google.com/javax/swing/MenuSelectionManager.html) manager)            Processes a mouse event forwarded from the MenuSelectionManager and changes the menu selection, if necessary, by using the MenuSelectionManager's API. |
| void | **JPopupMenu.**[**processMouseEvent**](http://docs.google.com/javax/swing/JPopupMenu.html#processMouseEvent(java.awt.event.MouseEvent,%20javax.swing.MenuElement%5B%5D,%20javax.swing.MenuSelectionManager))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event, [MenuElement](http://docs.google.com/javax/swing/MenuElement.html)[] path, [MenuSelectionManager](http://docs.google.com/javax/swing/MenuSelectionManager.html) manager)            This method is required to conform to the MenuElement interface, but it not implemented. |
| void | **JMenuBar.**[**processMouseEvent**](http://docs.google.com/javax/swing/JMenuBar.html#processMouseEvent(java.awt.event.MouseEvent,%20javax.swing.MenuElement%5B%5D,%20javax.swing.MenuSelectionManager))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event, [MenuElement](http://docs.google.com/javax/swing/MenuElement.html)[] path, [MenuSelectionManager](http://docs.google.com/javax/swing/MenuSelectionManager.html) manager)            Implemented to be a MenuElement -- does nothing. |
| void | **MenuElement.**[**processMouseEvent**](http://docs.google.com/javax/swing/MenuElement.html#processMouseEvent(java.awt.event.MouseEvent,%20javax.swing.MenuElement%5B%5D,%20javax.swing.MenuSelectionManager))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event, [MenuElement](http://docs.google.com/javax/swing/MenuElement.html)[] path, [MenuSelectionManager](http://docs.google.com/javax/swing/MenuSelectionManager.html) manager)            Processes a mouse event. |
| protected  void | **JComponent.**[**processMouseMotionEvent**](http://docs.google.com/javax/swing/JComponent.html#processMouseMotionEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Processes mouse motion events, such as MouseEvent.MOUSE\_DRAGGED. |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [javax.swing.event](http://docs.google.com/javax/swing/event/package-summary.html) | |
| --- | --- |

| Subclasses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [javax.swing.event](http://docs.google.com/javax/swing/event/package-summary.html) | |
| --- | --- |
| class | [**MenuDragMouseEvent**](http://docs.google.com/javax/swing/event/MenuDragMouseEvent.html)            MenuDragMouseEvent is used to notify interested parties that the menu element has received a MouseEvent forwarded to it under drag conditions. |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [javax.swing.plaf](http://docs.google.com/javax/swing/plaf/package-summary.html) | |
| --- | --- |

| Methods in [javax.swing.plaf](http://docs.google.com/javax/swing/plaf/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| boolean | **PopupMenuUI.**[**isPopupTrigger**](http://docs.google.com/javax/swing/plaf/PopupMenuUI.html#isPopupTrigger(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [javax.swing.plaf.basic](http://docs.google.com/javax/swing/plaf/basic/package-summary.html) | |
| --- | --- |

| Methods in [javax.swing.plaf.basic](http://docs.google.com/javax/swing/plaf/basic/package-summary.html) that return [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| protected  [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | **BasicComboPopup.**[**convertMouseEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.html#convertMouseEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |

| Methods in [javax.swing.plaf.basic](http://docs.google.com/javax/swing/plaf/basic/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| protected  void | **BasicSplitPaneDivider.DragController.**[**completeDrag**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.DragController.html#completeDrag(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Messages finishDraggingTo with the new location for the mouse event. |
| protected  void | **BasicSplitPaneDivider.DragController.**[**continueDrag**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.DragController.html#continueDrag(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Messages dragDividerTo with the new location for the mouse event. |
| protected  [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | **BasicComboPopup.**[**convertMouseEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.html#convertMouseEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| protected  void | **BasicComboPopup.**[**delegateFocus**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.html#delegateFocus(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            This is is a utility method that helps event handlers figure out where to send the focus when the popup is brought up. |
| protected  boolean | **BasicTreeUI.**[**isMultiSelectEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.html#isMultiSelectEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Returning true signifies a mouse event on the node should select from the anchor point. |
| boolean | **BasicPopupMenuUI.**[**isPopupTrigger**](http://docs.google.com/javax/swing/plaf/basic/BasicPopupMenuUI.html#isPopupTrigger(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| protected  boolean | **BasicTreeUI.**[**isToggleEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.html#isToggleEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Returning true indicates the row under the mouse should be toggled based on the event. |
| protected  boolean | **BasicTreeUI.**[**isToggleSelectionEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.html#isToggleSelectionEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Returning true signifies a mouse event on the node should toggle the selection of only the row under mouse. |
| void | **BasicToolBarUI.DockingListener.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicToolBarUI.DockingListener.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableHeaderUI.MouseInputHandler.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicTableHeaderUI.MouseInputHandler.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableUI.MouseInputHandler.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicTableUI.MouseInputHandler.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuUI.MouseInputHandler.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuUI.MouseInputHandler.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicListUI.MouseInputHandler.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicListUI.MouseInputHandler.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.BorderListener.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.BorderListener.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.GlassPaneDispatcher.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.GlassPaneDispatcher.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicFileChooserUI.DoubleClickListener.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicFileChooserUI.DoubleClickListener.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicButtonListener.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicButtonListener.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTreeUI.MouseInputHandler.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseInputHandler.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuItemUI.MouseInputHandler.**[**mouseClicked**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuItemUI.MouseInputHandler.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicToolBarUI.DockingListener.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicToolBarUI.DockingListener.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableHeaderUI.MouseInputHandler.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicTableHeaderUI.MouseInputHandler.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableUI.MouseInputHandler.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicTableUI.MouseInputHandler.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicSliderUI.TrackListener.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.TrackListener.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Set the models value to the position of the top/left of the thumb relative to the origin of the track. |
| void | **BasicScrollBarUI.TrackListener.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.TrackListener.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Set the models value to the position of the thumb's top of Vertical scrollbar, or the left/right of Horizontal scrollbar in left-to-right/right-to-left scrollbar relative to the origin of the track. |
| void | **BasicMenuUI.MouseInputHandler.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuUI.MouseInputHandler.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button is pressed on the menu and then dragged. |
| void | **BasicListUI.MouseInputHandler.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicListUI.MouseInputHandler.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.BorderListener.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.BorderListener.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.GlassPaneDispatcher.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.GlassPaneDispatcher.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicDesktopIconUI.MouseInputHandler.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopIconUI.MouseInputHandler.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicButtonListener.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicButtonListener.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTreeUI.MouseHandler.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseHandler.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTreeUI.MouseInputHandler.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseInputHandler.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuItemUI.MouseInputHandler.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuItemUI.MouseInputHandler.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicSplitPaneDivider.MouseHandler.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.MouseHandler.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            If dragger is not null it is messaged with continueDrag. |
| void | **BasicComboPopup.InvocationMouseMotionHandler.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.InvocationMouseMotionHandler.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicToolBarUI.DockingListener.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicToolBarUI.DockingListener.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableHeaderUI.MouseInputHandler.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicTableHeaderUI.MouseInputHandler.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableUI.MouseInputHandler.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicTableUI.MouseInputHandler.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuUI.MouseInputHandler.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuUI.MouseInputHandler.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the cursor enters the menu. |
| void | **BasicListUI.MouseInputHandler.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicListUI.MouseInputHandler.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.BorderListener.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.BorderListener.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.GlassPaneDispatcher.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.GlassPaneDispatcher.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicFileChooserUI.DoubleClickListener.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicFileChooserUI.DoubleClickListener.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            The JList used for representing the files is created by subclasses, but the selection is monitored in this class. |
| void | **BasicButtonListener.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicButtonListener.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTreeUI.MouseInputHandler.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseInputHandler.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuItemUI.MouseInputHandler.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuItemUI.MouseInputHandler.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicSplitPaneDivider.MouseHandler.**[**mouseEntered**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.MouseHandler.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse enters a component. |
| void | **BasicToolBarUI.DockingListener.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicToolBarUI.DockingListener.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableHeaderUI.MouseInputHandler.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicTableHeaderUI.MouseInputHandler.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableUI.MouseInputHandler.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicTableUI.MouseInputHandler.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicScrollBarUI.TrackListener.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.TrackListener.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse exits the scrollbar. |
| void | **BasicMenuUI.MouseInputHandler.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuUI.MouseInputHandler.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicListUI.MouseInputHandler.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicListUI.MouseInputHandler.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.BorderListener.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.BorderListener.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.GlassPaneDispatcher.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.GlassPaneDispatcher.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicButtonListener.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicButtonListener.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTreeUI.MouseInputHandler.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseInputHandler.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuItemUI.MouseInputHandler.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuItemUI.MouseInputHandler.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicSplitPaneDivider.MouseHandler.**[**mouseExited**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.MouseHandler.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse exits a component. |
| void | **BasicToolBarUI.DockingListener.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicToolBarUI.DockingListener.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableHeaderUI.MouseInputHandler.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicTableHeaderUI.MouseInputHandler.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableUI.MouseInputHandler.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicTableUI.MouseInputHandler.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicSliderUI.TrackListener.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.TrackListener.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicScrollBarUI.TrackListener.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.TrackListener.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuUI.MouseInputHandler.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuUI.MouseInputHandler.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicListUI.MouseInputHandler.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicListUI.MouseInputHandler.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.BorderListener.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.BorderListener.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.GlassPaneDispatcher.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.GlassPaneDispatcher.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicDesktopIconUI.MouseInputHandler.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopIconUI.MouseInputHandler.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicButtonListener.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicButtonListener.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTreeUI.MouseHandler.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseHandler.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse button has been moved on a component (with no buttons no down). |
| void | **BasicTreeUI.MouseInputHandler.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseInputHandler.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuItemUI.MouseInputHandler.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuItemUI.MouseInputHandler.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicSplitPaneDivider.MouseHandler.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.MouseHandler.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Resets the cursor based on the orientation. |
| void | **BasicComboPopup.ListMouseMotionHandler.**[**mouseMoved**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.ListMouseMotionHandler.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) anEvent) |
| void | **BasicToolBarUI.DockingListener.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicToolBarUI.DockingListener.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableHeaderUI.MouseInputHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicTableHeaderUI.MouseInputHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableUI.MouseInputHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicTableUI.MouseInputHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTabbedPaneUI.MouseHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicTabbedPaneUI.MouseHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicSliderUI.TrackListener.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.TrackListener.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            If the mouse is pressed above the "thumb" component then reduce the scrollbars value by one page ("page up"), otherwise increase it by one page. |
| void | **BasicScrollBarUI.TrackListener.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.TrackListener.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            If the mouse is pressed above the "thumb" component then reduce the scrollbars value by one page ("page up"), otherwise increase it by one page. |
| void | **BasicScrollBarUI.ArrowButtonListener.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.ArrowButtonListener.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuUI.MouseInputHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuUI.MouseInputHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse has been clicked on the menu. |
| void | **BasicListUI.MouseInputHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicListUI.MouseInputHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.BorderListener.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.BorderListener.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.GlassPaneDispatcher.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.GlassPaneDispatcher.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicDesktopIconUI.MouseInputHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopIconUI.MouseInputHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicButtonListener.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicButtonListener.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTreeUI.MouseHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when a mouse button has been pressed on a component. |
| void | **BasicTreeUI.MouseInputHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseInputHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuItemUI.MouseInputHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuItemUI.MouseInputHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicSplitPaneDivider.MouseHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.MouseHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Starts the dragging session by creating the appropriate instance of DragController. |
| void | **BasicComboPopup.InvocationMouseHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.InvocationMouseHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Responds to mouse-pressed events on the combo box. |
| void | **BasicComboPopup.ListMouseHandler.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.ListMouseHandler.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicToolBarUI.DockingListener.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicToolBarUI.DockingListener.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableHeaderUI.MouseInputHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicTableHeaderUI.MouseInputHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTableUI.MouseInputHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicTableUI.MouseInputHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicSliderUI.TrackListener.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicSliderUI.TrackListener.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicScrollBarUI.TrackListener.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.TrackListener.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicScrollBarUI.ArrowButtonListener.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicScrollBarUI.ArrowButtonListener.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuUI.MouseInputHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuUI.MouseInputHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Invoked when the mouse has been released on the menu. |
| void | **BasicListUI.MouseInputHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicListUI.MouseInputHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.BorderListener.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.BorderListener.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicInternalFrameUI.GlassPaneDispatcher.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicInternalFrameUI.GlassPaneDispatcher.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicDesktopIconUI.MouseInputHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicDesktopIconUI.MouseInputHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicButtonListener.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicButtonListener.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTreeUI.MouseHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicTreeUI.MouseInputHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseInputHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicMenuItemUI.MouseInputHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicMenuItemUI.MouseInputHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **BasicSplitPaneDivider.MouseHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.MouseHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            If dragger is not null it is messaged with completeDrag. |
| void | **BasicComboPopup.InvocationMouseHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.InvocationMouseHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Responds to the user terminating a click or drag that began on the combo box. |
| void | **BasicComboPopup.ListMouseHandler.**[**mouseReleased**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.ListMouseHandler.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) anEvent) |
| protected  int | **BasicSplitPaneDivider.DragController.**[**positionForMouseEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.DragController.html#positionForMouseEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Returns the new position to put the divider at based on the passed in MouseEvent. |
| protected  int | **BasicSplitPaneDivider.VerticalDragController.**[**positionForMouseEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.VerticalDragController.html#positionForMouseEvent(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Returns the new position to put the divider at based on the passed in MouseEvent. |
| void | **BasicRadioButtonMenuItemUI.**[**processMouseEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicRadioButtonMenuItemUI.html#processMouseEvent(javax.swing.JMenuItem,%20java.awt.event.MouseEvent,%20javax.swing.MenuElement%5B%5D,%20javax.swing.MenuSelectionManager))([JMenuItem](http://docs.google.com/javax/swing/JMenuItem.html) item, [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e, [MenuElement](http://docs.google.com/javax/swing/MenuElement.html)[] path, [MenuSelectionManager](http://docs.google.com/javax/swing/MenuSelectionManager.html) manager) |
| void | **BasicCheckBoxMenuItemUI.**[**processMouseEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicCheckBoxMenuItemUI.html#processMouseEvent(javax.swing.JMenuItem,%20java.awt.event.MouseEvent,%20javax.swing.MenuElement%5B%5D,%20javax.swing.MenuSelectionManager))([JMenuItem](http://docs.google.com/javax/swing/JMenuItem.html) item, [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e, [MenuElement](http://docs.google.com/javax/swing/MenuElement.html)[] path, [MenuSelectionManager](http://docs.google.com/javax/swing/MenuSelectionManager.html) manager) |
| protected  void | **BasicTreeUI.**[**selectPathForEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.html#selectPathForEvent(javax.swing.tree.TreePath,%20java.awt.event.MouseEvent))([TreePath](http://docs.google.com/javax/swing/tree/TreePath.html) path, [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Messaged to update the selection based on a MouseEvent over a particular row. |
| protected  boolean | **BasicTreeUI.**[**startEditing**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.html#startEditing(javax.swing.tree.TreePath,%20java.awt.event.MouseEvent))([TreePath](http://docs.google.com/javax/swing/tree/TreePath.html) path, [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Will start editing for node if there is a cellEditor and shouldSelectCell returns true. |
| protected  void | **BasicComboPopup.**[**updateListBoxSelectionForEvent**](http://docs.google.com/javax/swing/plaf/basic/BasicComboPopup.html#updateListBoxSelectionForEvent(java.awt.event.MouseEvent,%20boolean))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) anEvent, boolean shouldScroll)            A utility method used by the event listeners. |

| Constructors in [javax.swing.plaf.basic](http://docs.google.com/javax/swing/plaf/basic/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| [**BasicSplitPaneDivider.DragController**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.DragController.html#BasicSplitPaneDivider.DragController(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| [**BasicSplitPaneDivider.VerticalDragController**](http://docs.google.com/javax/swing/plaf/basic/BasicSplitPaneDivider.VerticalDragController.html#BasicSplitPaneDivider.VerticalDragController(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| [**BasicTreeUI.MouseInputHandler**](http://docs.google.com/javax/swing/plaf/basic/BasicTreeUI.MouseInputHandler.html#BasicTreeUI.MouseInputHandler(java.awt.Component,%20java.awt.Component,%20java.awt.event.MouseEvent))([Component](http://docs.google.com/java/awt/Component.html) source, [Component](http://docs.google.com/java/awt/Component.html) destination, [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event) |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [javax.swing.plaf.metal](http://docs.google.com/javax/swing/plaf/metal/package-summary.html) | |
| --- | --- |

| Methods in [javax.swing.plaf.metal](http://docs.google.com/javax/swing/plaf/metal/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| void | **MetalComboBoxUI.MetalComboPopup.**[**delegateFocus**](http://docs.google.com/javax/swing/plaf/metal/MetalComboBoxUI.MetalComboPopup.html#delegateFocus(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)  **Deprecated.** |
| void | **MetalToolBarUI.MetalDockingListener.**[**mouseDragged**](http://docs.google.com/javax/swing/plaf/metal/MetalToolBarUI.MetalDockingListener.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **MetalToolBarUI.MetalDockingListener.**[**mousePressed**](http://docs.google.com/javax/swing/plaf/metal/MetalToolBarUI.MetalDockingListener.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [javax.swing.plaf.multi](http://docs.google.com/javax/swing/plaf/multi/package-summary.html) | |
| --- | --- |

| Methods in [javax.swing.plaf.multi](http://docs.google.com/javax/swing/plaf/multi/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| boolean | **MultiPopupMenuUI.**[**isPopupTrigger**](http://docs.google.com/javax/swing/plaf/multi/MultiPopupMenuUI.html#isPopupTrigger(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) a)            Invokes the isPopupTrigger method on each UI handled by this object. |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [javax.swing.table](http://docs.google.com/javax/swing/table/package-summary.html) | |
| --- | --- |

| Methods in [javax.swing.table](http://docs.google.com/javax/swing/table/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| [String](http://docs.google.com/java/lang/String.html) | **JTableHeader.**[**getToolTipText**](http://docs.google.com/javax/swing/table/JTableHeader.html#getToolTipText(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Allows the renderer's tips to be used if there is text set. |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) | |
| --- | --- |

| Methods in [javax.swing.text](http://docs.google.com/javax/swing/text/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| [String](http://docs.google.com/java/lang/String.html) | **JTextComponent.**[**getToolTipText**](http://docs.google.com/javax/swing/text/JTextComponent.html#getToolTipText(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) event)            Returns the string to be used as the tooltip for event. |
| void | **DefaultCaret.**[**mouseClicked**](http://docs.google.com/javax/swing/text/DefaultCaret.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Called when the mouse is clicked. |
| void | **DefaultCaret.**[**mouseDragged**](http://docs.google.com/javax/swing/text/DefaultCaret.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Moves the caret position according to the mouse pointer's current location. |
| void | **DefaultCaret.**[**mouseEntered**](http://docs.google.com/javax/swing/text/DefaultCaret.html#mouseEntered(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Called when the mouse enters a region. |
| void | **DefaultCaret.**[**mouseExited**](http://docs.google.com/javax/swing/text/DefaultCaret.html#mouseExited(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Called when the mouse exits a region. |
| void | **DefaultCaret.**[**mouseMoved**](http://docs.google.com/javax/swing/text/DefaultCaret.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Called when the mouse is moved. |
| void | **DefaultCaret.**[**mousePressed**](http://docs.google.com/javax/swing/text/DefaultCaret.html#mousePressed(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            If button 1 is pressed, this is implemented to request focus on the associated text component, and to set the caret position. |
| void | **DefaultCaret.**[**mouseReleased**](http://docs.google.com/javax/swing/text/DefaultCaret.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Called when the mouse is released. |
| protected  void | **DefaultCaret.**[**moveCaret**](http://docs.google.com/javax/swing/text/DefaultCaret.html#moveCaret(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Tries to move the position of the caret from the coordinates of a mouse event, using viewToModel(). |
| protected  void | **DefaultCaret.**[**positionCaret**](http://docs.google.com/javax/swing/text/DefaultCaret.html#positionCaret(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Tries to set the position of the caret from the coordinates of a mouse event, using viewToModel(). |

| Uses of [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) in [javax.swing.text.html](http://docs.google.com/javax/swing/text/html/package-summary.html) | |
| --- | --- |

| Methods in [javax.swing.text.html](http://docs.google.com/javax/swing/text/html/package-summary.html) with parameters of type [MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) | |
| --- | --- |
| void | **HTMLEditorKit.LinkController.**[**mouseClicked**](http://docs.google.com/javax/swing/text/html/HTMLEditorKit.LinkController.html#mouseClicked(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e)            Called for a mouse click event. |
| void | **HTMLEditorKit.LinkController.**[**mouseDragged**](http://docs.google.com/javax/swing/text/html/HTMLEditorKit.LinkController.html#mouseDragged(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **HTMLEditorKit.LinkController.**[**mouseMoved**](http://docs.google.com/javax/swing/text/html/HTMLEditorKit.LinkController.html#mouseMoved(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) e) |
| void | **FormView.MouseEventListener.**[**mouseReleased**](http://docs.google.com/javax/swing/text/html/FormView.MouseEventListener.html#mouseReleased(java.awt.event.MouseEvent))([MouseEvent](http://docs.google.com/java/awt/event/MouseEvent.html) evt) |

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | [**Class**](http://docs.google.com/java/awt/event/MouseEvent.html) | **Use** | [**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   NEXT | [**FRAMES**](http://docs.google.com/index.html?java/awt/event//class-useMouseEvent.html)    [**NO FRAMES**](http://docs.google.com/MouseEvent.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[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).