Listener Interfaces

Listener	Methods on the Interface	Added method	Associate Components
ActionListener	actionPerformed(ActionEvent)	addActionListener()	JButton, JCheckBox, JComboBox, JTextField, JRadioButton, JMenuItem

ItemListener	focusGained(FocusEvent)
	focusLost(FocusEvent)
		addFocusListener()	All Swing components

ItemListener	itemStateChanged(ItemEvent)	addItemListener()	JButton, JCheckBox, JComboBox, JRadioButton components

KeyListener	keyPressed(KeyEvent)
	keyReleased(KeyEvent)
	keyTyped(KeyEvent)
		addKeyListener()	All Swing components

MouseListener	mouseClicked(MouseEvent)
	mouseEntered(MouseEvent)
	mouseExited(MouseEvent)
	mousePressed(MouseEvent)
	mouseReleased(MouseEvent)
		addMouseListener()	All Swing components

MouseMotionListener	mouseDragged(MouseEvent)
	mouseMoved(MouseEvent)
		addMouseMotionListener	All Swing Components

WindowListener	windowActivated(WindowEvent)
	windowClosed(WindowEvent)
	windowDeactivated(WindowEvent)
	windowDeiconified(WindowEvent)
	windowIconified(WindowEvent)
	windowOpened(WindowEvent)
		addWindowListener()	JWindow, JFrame

Adapter classes

MouseAdapter (Abstract Classe)	mouseClicked(MouseEvente)
	mouseEntered(MouseEvente)
	mouseExited(MouseEvent)
	mousePressed(MouseEvent)
	mouseReleased(MouseEvent)
		addMouseListener()	

KeyAdapter (Abstract Classe)	keyPressed(KeyEvent)
	keyReleased(KeyEvent)
	keyTyped(KeyEvent)
		addKeyListener()	

WindowAdapter (Abstract class)	windowActivated(WindowEvent)
	windowClosed(WindowEvent)
	windowClosing(WindowEvent
	windowDeactivated(WindowEvent)
	windowDeiconified(WindowEvent)
		addWindowListener()	

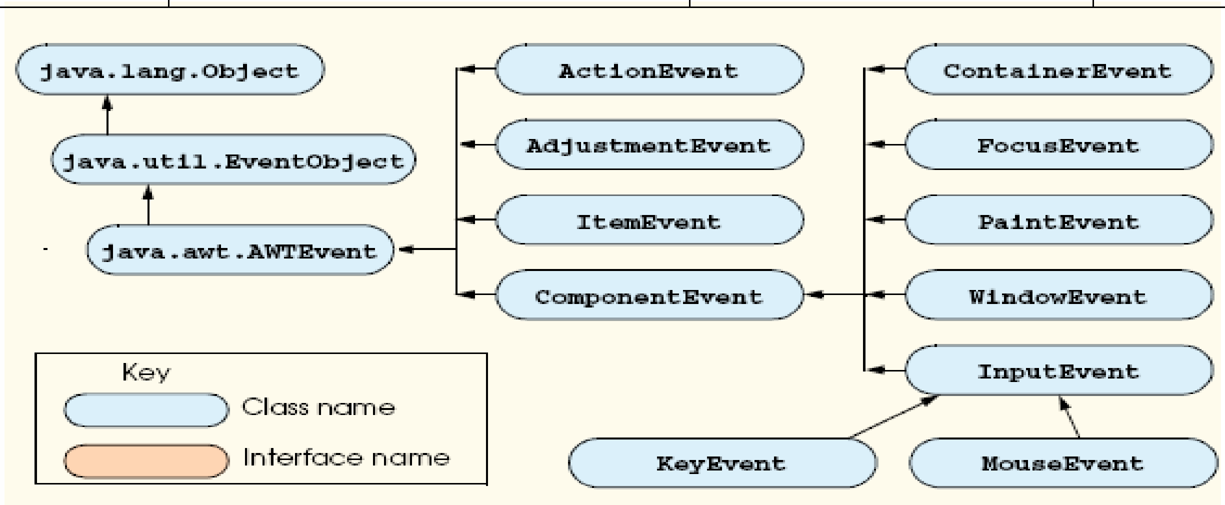
	windowGainedFocus(WindowEvent)
	windowIconified(WindowEvent)
	windowLostFocus(WindowEvent)
	windowOpened(WindowEvent)
	windowStateChanged(WindowEvent)		

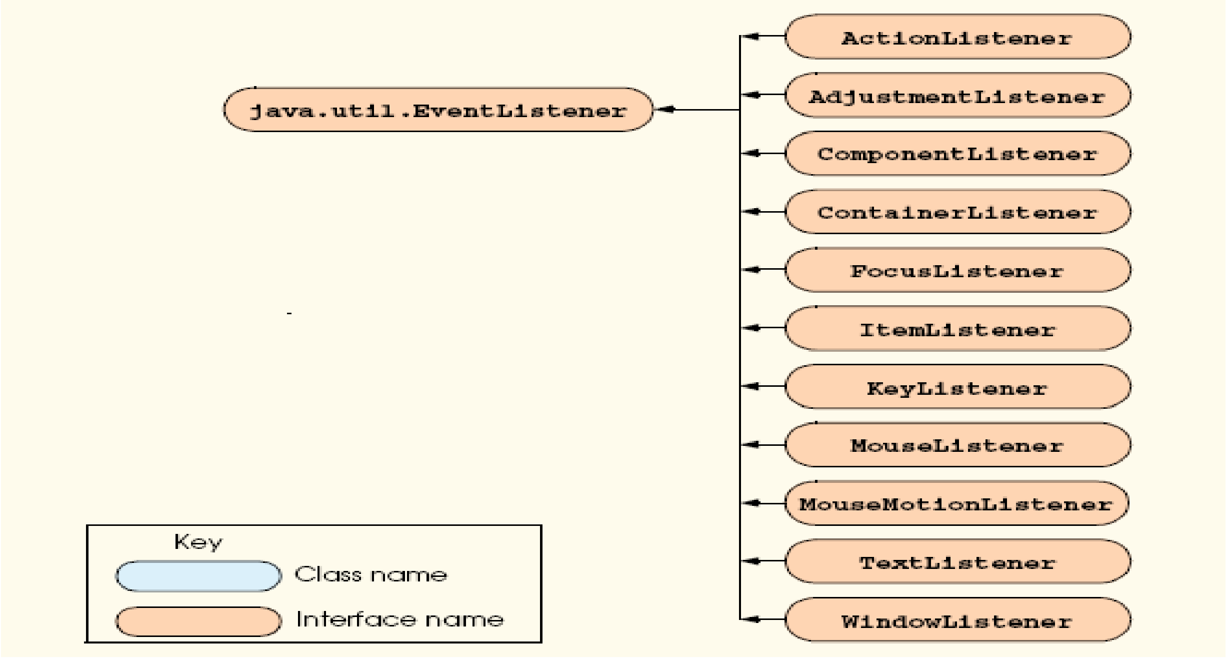
FocusAdapter	focusGained(FocusEvent)
	focusLost(FocusEvent)
		addFocusListener()	All Swing Components

MouseMotionAdapter	mouseDragged(MouseEvent)
	mouseMoved(MouseEvent)
		addMouseMotionListener()	

ContainerAdapter	componentAdded(ContainerEvent)
	componentRemoved(ContainerEvent)
		addContainerListener()	

ComponentAdapter	componentHidden(ComponentEvent)
	componentMoved(ComponentEvent)
	componentResized(ComponentEvent)
	componentShown(ComponentEvent)
		addComponentListener()	





All Listener Interfaces, Event Classes & Adapter Classes JAVA-5.0

Interface Summary	
ActionListener	The listener interface for receiving action events.
AdjustmentListener	The listener interface for receiving adjustment events.
AWTEventListener	The listener interface for receiving notification of events dispatched to objects that are instances of Component or MenuComponent or their subclasses.
ComponentListener	The listener interface for receiving component events.
ContainerListener	The listener interface for receiving container events.
FocusListener	The listener interface for receiving keyboard focus events on a component.
HierarchyBoundsListener	The listener interface for receiving ancestor moved and resized events.
HierarchyListener	The listener interface for receiving hierarchy changed events.
InputMethodListener	The listener interface for receiving input method events.
ItemListener	The listener interface for receiving item events.
KeyListener	The listener interface for receiving keyboard events (keystrokes).
MouseListener	The listener interface for receiving "interesting" mouse events (press, release, click, enter, and exit) on a component.
MouseMotionListener	The listener interface for receiving mouse motion events on a component.
MouseWheelListener	The listener interface for receiving mouse wheel events on a component.
TextListener	The listener interface for receiving text events.
WindowFocusListener	The listener interface for receiving WindowEvents, including WINDOW_GAINED_FOCUS and WINDOW_LOST_FOCUS events.
WindowListener	The listener interface for receiving window events.
WindowStateListener	The listener interface for receiving window state events.

Class Summary	
ActionEvent	A semantic event which indicates that a component-defined action occurred.
AdjustmentEvent	The adjustment event emitted by Adjustable objects.
AWTEventListenerProxy	A class which extends the EventListenerProxy, specifically for adding an AWTEventListener for a specific event mask.
ComponentAdapter	An abstract adapter class for receiving component events.
ComponentEvent	A low-level event which indicates that a component moved, changed size, or changed visibility (also, the root class for the other component-level events).
ContainerAdapter	An abstract adapter class for receiving container events.
ContainerEvent	A low-level event which indicates that a container's contents changed because a component was added or removed.
FocusAdapter	An abstract adapter class for receiving keyboard focus events.
FocusEvent	A low-level event which indicates that a Component has gained or lost the input focus.
HierarchyBoundsAdapter	An abstract adapter class for receiving ancestor moved and resized events.
HierarchyEvent	An event which indicates a change to the Component hierarchy to which a Component belongs.
InputEvent	The root event class for all component-level input events.
InputMethodEvent	Input method events contain information about text that is being composed using an input method.
InvocationEvent	An event which executes the run() method on a Runnable when dispatched by the AWT event dispatcher thread.
ItemEvent	A semantic event which indicates that an item was selected or deselected.
KeyAdapter	An abstract adapter class for receiving keyboard events.
KeyEvent	An event which indicates that a keystroke occurred in a component.
MouseAdapter	An abstract adapter class for receiving mouse events.
MouseEvent	An event which indicates that a mouse action occurred in a component.
MouseMotionAdapter	An abstract adapter class for receiving mouse motion events.
MouseWheelEvent	An event which indicates that the mouse wheel was rotated in a component.
PaintEvent	The component-level paint event.
TextEvent	A semantic event which indicates that an object's text changed.
WindowAdapter	An abstract adapter class for receiving window events.
WindowEvent	A low-level event that indicates that a window has changed its status.

Class Event

static int	ACTION_EVENT	This event indicates that the user wants some action to occur.
static int	ALT_MASK	This flag indicates that the Alt key was down when the event occurred.
Object	arg	An arbitrary argument of the event.
static int	BACK_SPACE	The BackSpace key.
static int	CAPS_LOCK	The Caps Lock key, a non-ASCII action key.
int	clickCount	For MOUSE_DOWN events, this field indicates the number of consecutive clic
static int	CTRL_MASK	This flag indicates that the Control key was down when the event occurred.
static int	DELETE	The Delete key.
static int	DOWN	The Down Arrow key, a non-ASCII action key.
static int	END	The End key, a non-ASCII action key.
static int	ENTER	The Enter key.
static int	ESCAPE	The Escape key.
Event	evt	The next event.
static int	F1	The F1 function key, a non-ASCII action key.
static int	F10	The F10 function key, a non-ASCII action key.
static int	F11	The F11 function key, a non-ASCII action key.
static int	F12	The F12 function key, a non-ASCII action key.
static int	F2	The F2 function key, a non-ASCII action key.
static int	F3	The F3 function key, a non-ASCII action key.
static int	F4	The F4 function key, a non-ASCII action key.
static int	F5	The F5 function key, a non-ASCII action key.
static int	F6	The F6 function key, a non-ASCII action key.
static int	F7	The F7 function key, a non-ASCII action key.
static int	F8	The F8 function key, a non-ASCII action key.
static int	F9	The F9 function key, a non-ASCII action key.
static int	GOT_FOCUS	A component gained the focus.
static int	HOME	The Home key, a non-ASCII action key.
int	id	Indicates which type of event the event is, and which other Event variables are relevant
static int	INSERT	The Insert key, a non-ASCII action key.
int	key	The key code of the key that was pressed in a keyboard event.
static int	KEY_ACTION	The user has pressed a non-ASCII *action* key.
static int	KEY_ACTION_RELEASE	The user has released a non-ASCII *action* key.
static int	KEY_PRESS	The user has pressed a normal key.
static int	KEY_RELEASE	The user has released a normal key.
static int	LEFT	The Left Arrow key, a non-ASCII action key.
static int	LIST_DESELECT	An item in a list has been deselected.

static int	LIST_SELECT An item in a list has been selected.
static int	LOAD_FILE A file loading event.
static int	LOST_FOCUS A component lost the focus.
static int	META_MASK This flag indicates that the Meta key was down when the event occurred.
int	modifiers The state of the modifier keys.
static int	MOUSE_DOWN The user has pressed the mouse button.
static int	MOUSE_DRAG The user has moved the mouse with a button pressed.
static int	MOUSE_ENTER The mouse has entered a component.
static int	MOUSE_EXIT The mouse has exited a component.
static int	MOUSE_MOVE The mouse has moved with no button pressed.
static int	MOUSE_UP The user has released the mouse button.
static int	NUM_LOCK The Num Lock key, a non-ASCII action key.
static int	PAUSE The Pause key, a non-ASCII action key.
static int	PGDN The Page Down key, a non-ASCII action key.
static int	PGUP The Page Up key, a non-ASCII action key.
static int	PRINT_SCREEN The Print Screen key, a non-ASCII action key.
static int	RIGHT The Right Arrow key, a non-ASCII action key.
static int	SAVE_FILE A file saving event.
static int	SCROLL_BEGIN The scroll begin event.
static int	SCROLL_END The scroll end event.
static int	SCROLL_LINE_DOWN The user has activated the *line down* area of a scroll bar.
static int	SCROLL_LINE_UP The user has activated the *line up* area of a scroll bar.
static int	SCROLL_LOCK The Scroll Lock key, a non-ASCII action key.
static int	SCROLL_PAGE_DOWN The user has activated the *page down* area of a scroll bar.
static int	SCROLL_PAGE_UP The user has activated the *page up* area of a scroll bar.
static int	SHIFT_MASK This flag indicates that the Shift key was down when the event occurred.
static int	TAB The Tab key.
Object	target The target component.
static int	UP The Up Arrow key, a non-ASCII action key.
long	when The time stamp.
static int	WINDOW_DEICONIFY The user has asked the window manager to de-iconify the window.
static int	WINDOW_DESTROY The user has asked the window manager to kill the window.
static int	WINDOW_EXPOSE The user has asked the window manager to expose the window.
static int	WINDOW_ICONIFY The user has asked the window manager to iconify the window.
static int	WINDOW_MOVED The user has asked the window manager to move the window.
int	x The *x* coordinate of the event.
int	y The *y* coordinate of the event.