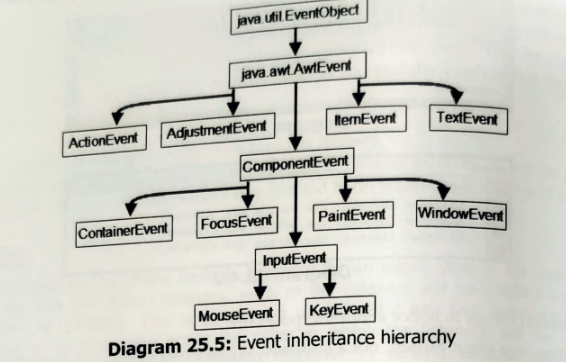
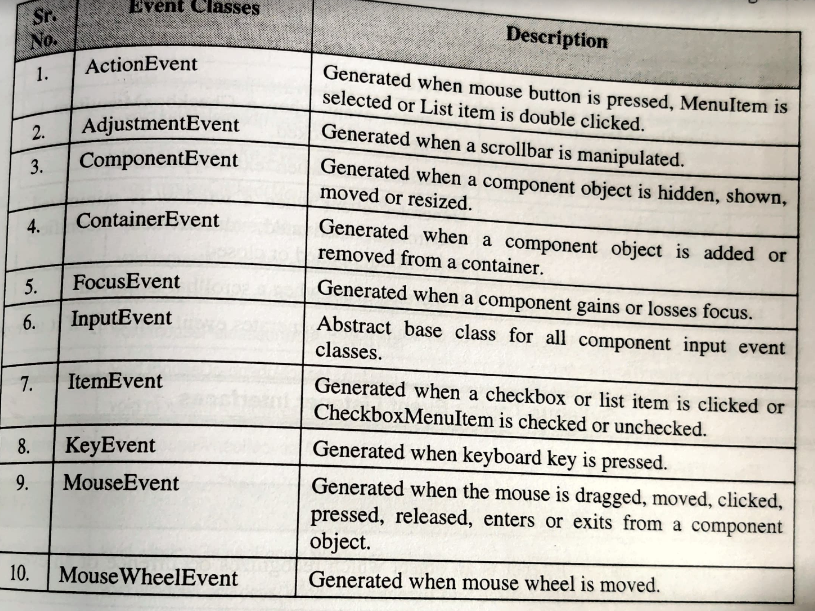
**Event Handling**

* The mechanism of delegation event model is: the source generates an event and sends it to multiple listeners.
* **Event is an action performed on any source.** Event is a result of a user's action which changes the state of a source. Example, selecting an item from a list, checking or unchecking a checkbox, etc.
* **Source is the component object on which action is performed.** A source can generate more than one type of event.
* Each event has its own registration method which registers listeners.
* For example, addKeyListener() registers keyboard keys with event listener, addActionListener() registers any select action with event listener.
* **Listener is the object which gets notified when an event occurs**. Example, MouseListener interface defines methods related to mouse clicked, pressed, released, dragged, moved, entered, exit. KeyListener interface defines methods related to keyboard’s pressed, released and typed.



**Event Classes**

* Event classes are at the core of Java’s event handling mechanism. All the listeners and methods are encapsulated with the event class.
* All the event classes are present under **java.awt.event packages.**

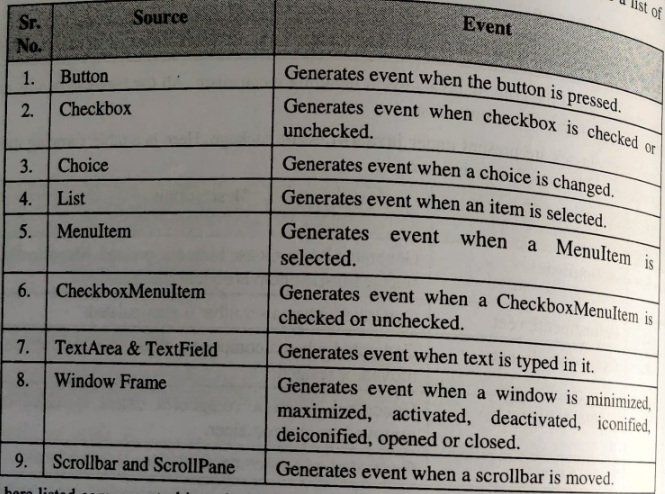


11. TextEvent → Generated when the value of a TextArea or TextField is changed.

12. WindowEvent → Generated when a window is activated, closed, opened, iconified, deactivated or about to be closed.

These all classes are responsible for all events and an essential ingredient in all listeners.

**Event Sources**



**Event Interfaces**

