Event Handling

Event

- **Object that describes a state change in source**
- **e.g Pressing a Button generates Action Event [Button is source of Event, Name of the event is Action]**
- **Every event has a source whose state change causes the event**
- **Some of the activities that may result in event generation:**

1. **Pressing a Button**
2. **Entering a Character from keyboard**
3. **Selecting an item in a list**
4. **Clicking the mouse**
5. **Timer expires**
6. **Counter Exceeds**

Event Source

- Source is an object that generates an event
- Internal state of the source changes in some way
- When ever source generates an event it is notified to one or more listeners
- Source must register one or more listeners via following methods:

1. public void addTypeListener(TypeListener el)
2. public void addTypeListener(TypeListener el) throws java.util.TooManyListnersException

- Second Form allows only single listener to attached to the source.
- <<Type>> is the type of the Event

Event Listeners

- Object That is notified whenever source generates a particular type of event
- Responsibilities

1. Must register with the source
2. Must implement desired Listener interface

Delegation Event Model

- Modern Approach for event handling
- Three components

1. Source [Object which generates the vent]
2. Event [Object that indicates state change in Source]
3. Listener [Notified Object which carries out some action whenever event is generated]

- Source generates a particular type of event and it notifies the listener
- On receiving notification from source listener carries out the desired task

Examples of Events

1. **ActionEvent**

   (a) When Button is Pressed

   (b) When List Item is double-clicked

   (c) Menu Item is selected

2. **AdjustmentEvent**

   (a) Scroll Bar manipulated

3. **ComponetEvent**

   (a) Component is hidden, moved, resized or becomes visible

4. **ContainerEvent**

   (a) Whenever a component is added or removed from conatiner

5. **FocusEvent**

   (a) When ever component gains or lose keyboard focus

6 Mouse Event

(a) Mouse Dragged , moved, clicked, pressed or released

Event Sources Examples

- User interface components that can generate events

1.	Button	(ActionEvent)
2.	Checkbox	(ItemEvent)
3.	Choice	(ItemEvent)
4.	List	(ActionEvent, ItemEvent)
5.	MenuItem	(ActionEvent, ItemEvent)
6.	Scrollbar	(AdjustmentEvent)
7.	Window	(WindowEvent)

Event Classes

- EventObject (Super Class for for all events, java.util.*)
- AWTEvent (Subclass of EventObject, Superclass of all AWT based events)
- Package java.awt.event defines event classes
- Examples : ActionEvent, ContainerEvent, FocusEvent

Event Listener Interfaces

- **Listener must implement suitable event listener interface in order to execute the code whenever that type of event is generated**
- **What happens when an event occus?**

1. **Suitable Event object is created**
2. **Source invokes the appropriate method provided by the listener by passing the object created in step1 as parameter**

- **Examples : ActionListener, MouseListner, MouseMotionListener**

ActionListener Interface

```
public interface ActionListener
{
public void actionPerformed(ActionEvent ae);
}
```

MouseListener

```
public interface MouseListener
{
public void mouseClicked(MouseEvent me);
public void mouseEntered(MouseEvent me);
public void mouseExited(MouseEvent me);
public void mousePressed(MouseEvent me);
public void mouseReleased(MouseEvent me);
}
```

MouseMotionListener extends MouseListener

```
public interface MouseListener extends MouseMotionListener
{
public void mouseMoved(MouseEvent me);
public void mouseDragged(MouseEvent me);
}
```