Event Listener interfaces (Java Event Handling)

Changing the state of an object is known as an event. For example, click on button, dragging mouse etc. The event delegation model contains two main components. First are the event sources and second are the listeners. Most of the listener interfaces are available in the *java.awt.event* package. which are listed below:

Java Event classes and Listener interfaces

Event Classes	Listener Interfaces
ActionEvent	ActionListener
MouseEvent	MouseListener and MouseMotionListener
MouseWheelEvent	MouseWheelListener
KeyEvent	KeyListener
ItemEvent	ItemListener
TextEvent	TextListener
AdjustmentEvent	AdjustmentListener
WindowEvent	WindowListener
ComponentEvent	ComponentListener

ContainerEvent	ContainerListener
FocusEvent	FocusListener

Steps to perform Event Handling

Following steps are required to perform event handling:

1. Register the component with the Listener

1.Registration Methods

For registering the component with the Listener, many classes provide the registration methods. For example:

- **Button**
    - public void addActionListener(ActionListener a){}
- **MenuItem**
    - public void addActionListener(ActionListener a){}
- **TextField**
    - public void addActionListener(ActionListener a){}
    - public void addTextListener(TextListener a){}
- **TextArea**
    - public void addTextListener(TextListener a){}
- **Checkbox**
    - public void addItemListener(ItemListener a){}
- **Choice**
    - public void addItemListener(ItemListener a){}

2.Java event handling by implementing ActionListener

We can put the event handling code into one of the following places:

1. Within class

2. Other class

3. Anonymous class

Following is an example where we written code for event handling withen

```
1.  import java.awt.*;
2.  import java.awt.event.*;
3.  class AEvent extends Frame implements ActionListener{
4.  TextField tf;
5.  AEvent(){
6.
7.  //create components
8.  tf=new TextField();
9.  tf.setBounds(60,50,170,20);
10  Button b=new Button("click me");
11  b.setBounds(100,120,80,30);
12
13  //register listener
14  b.addActionListener(this);//passing current instance
15
16  //add components and set size, layout and visibility
17  add(b);add(tf);
18  setSize(300,300);
19  setLayout(null);
20  setVisible(true);
21  }
22  public void actionPerformed(ActionEvent e){
23  tf.setText("Welcome");
24  }
25  public static void main(String args[]){
26  new AEvent();
27  }
28  }
```

public void setBounds(int xaxis, int yaxis, int width, int height); have been used in the above example that sets the position of the component it may be button, textfield etc.