AWT Event Handling Programs

Program 1- Count No of words and characters

```
import java.awt.*;
import java.net.*;
import java.awt.event.*;

public class LearnAWT extends Frame {
        TextArea ta;
        Label l1;
        Label l2;
        Button b;

        LearnAWT() {
          setTitle("Word Counter");
          ta = new TextArea();
          ta.setBounds(100, 100, 400,
400);
          b = new Button("Count");
          b.setBounds(270,500,60,40);
          l1 = new Label("");
          l1.setBounds(100, 50, 100,
30);
          l2 = new Label("");
          l2.setBounds(250,50,100,30);
          add(b);
          add(ta);
          add(l1);
          add(l2);
          setLayout(null);
          setSize(600,600);
          setVisible(true);

          b.addActionListener(new
ActionListener(){
          public void
actionPerformed(ActionEvent e) {
            int s =
ta.getText().split("\\s+").length;
            int c = ta.getText().length();
            l1.setText("Words " + s);
            l2.setText("Characters " + c);
          }
        });
  }
public static void main(String []args) {
    new LearnAWT();
  }
}
```

Program 2 – Find and Replace

```
import java.awt.*;
import java.awt.event.*;

public class FindReplace extends
Frame
{
TextArea textArea = new TextArea(8,
40);
TextField from= new TextField(8);
TextField to= new TextField(8);

  public FindReplace()
{
   setTitle("Find and Replace");
   setSize(700, 300);
   setVisible(true);
   setLayout(new FlowLayout());
   addWindowListener(new
WindowAdapter() {
     public void
windowClosing(WindowEvent e) {
       System.exit(0);
     }
   });

   Panel p = new Panel();
   Button replace = new
Button("Replace");
   p.add(textArea);

   replace.addActionListener(new
ActionListener() {
      public void
actionPerformed(ActionEvent evt) {
        String f = from.getText();
        int start =
textArea.getText().indexOf(f);
        if (start >= 0 && f.length() > 0)

textArea.replaceRange(to.getText(),
start, start+ f.length());
     }
   });
   p.add(new Label("Find"));
   p.add(from);
   p.add(new Label("Replace"));
   p.add(to);
   p.add(replace);
   add(p);
```

```
  }

  public static void main(String[] args) {
    FindReplace f = new FindReplace();

  }
}
```

Program 3 – Item Event Example

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*<applet code="ColorChoice"
width=300 height=300></applet>*/
public class ColorChoice extends Applet
implements ItemListener
 {
  Choice c1;
  Color color;

  public void init()
{
    c1 = new Choice();
    c1.addItem("Red");
    c1.addItem("Green");
    c1.addItem("Blue");
    c1.addItemListener(this);
    add(c1);

  }
public void
itemStateChanged(ItemEvent ie)
  {

    if(c1.getSelectedItem().equals("Red"))
              color = Color.red;
    else
if(c1.getSelectedItem().equals("Green"))
              color = Color.green;
    else
              color = Color.blue;

    repaint();
  }
  public void paint(Graphics g)
  {
      setBackground(color);
  }
}
```

Program 4 – Action Event with TextField Example

```
import java.awt.*;
import java.awt.event.*;
class AEvent extends Frame
implements ActionListener{
TextField tf;
AEvent(){
  //create components
tf=new TextField();
tf.setBounds(60,50,170,20);
Button b=new Button("click me");
b.setBounds(100,120,80,30);

//register listener
b.addActionListener(this);//passing
current instance

//add components and set size, layout
and visibility
add(b);add(tf);
setSize(300,300);
setLayout(null);
setVisible(true);
}
public void
actionPerformed(ActionEvent e)
{
tf.setText("Welcome");
}
public static void main(String args[])
{
new AEvent();
}
}
```

Program 5 – Adjustment Event with ScrollBar Example

```
import java.awt.*;
import java.awt.event.*;

class AdjustmentEventEx extends
Frame implements AdjustmentListener
{
        int rval=0,gval=0,bval=0;
        Scrollbar sr,sb,sg;
        Panel ps;
        AdjustmentEventEx()
```

```
        {
                ps=new Panel();
                Label l1=new Label("Red");
                Label l2=new
Label("Green");
                Label l3=new
Label("Blue");
                sr=new
Scrollbar(Scrollbar.HORIZONTAL,0,5,0,
255);
                sg=new
Scrollbar(Scrollbar.HORIZONTAL,0,5,0,
255);
                sb=new
Scrollbar(Scrollbar.HORIZONTAL,0,5,0,
255);
                ps.add(l1);
                ps.add(sr);
                ps.add(l2);
                ps.add(sg);
                ps.add(l3);
                ps.add(sb);
                add("South",ps);

        sr.addAdjustmentListener(this);

        sg.addAdjustmentListener(this);

sb.addAdjustmentListener(this);
                setSize(400,400);
                setVisible(true);
        }
        public void
adjustmentValueChanged(
AdjustmentEvent e)
        {
                if(e.getSource().equals(sr))
                {
                        rval=sr.getValue();
                        setBackground(new
Color(rval,bval,gval));
                }
                if(e.getSource().equals(sg))
                {
                        gval=sg.getValue();
                        setBackground(new
Color(rval,gval,bval));
                }
                if(e.getSource().equals(sb))
                {
                        bval=sb.getValue();
                        setBackground(new
Color(rval,gval,bval));
                }
        }

        public static void main(String[]
args)
        {
                AdjustmentEventEx
ae=new AdjustmentEventEx();
        }
}
```

Program 6 – Action Event with Menu Example

```
import java.awt.*;
import java.awt.event.*;

public class SimpleMenuExample
extends Frame implements
ActionListener
{
  Menu states, cities;
TextArea ta;
  public SimpleMenuExample()
  {
ta = new TextArea(10, 40);
    ta.setBackground(Color.cyan);

    MenuBar mb = new MenuBar();
                // begin with creating
menu bar
    setMenuBar(mb);
        // add menu bar to frame

    states = new Menu("Indian States");
        // create menus
    cities = new Menu("Indian Cities");

    mb.add(states);
        // add menus to menu bar
    mb.add(cities);

    states.addActionListener(this);
        // link with ActionListener for
event handling
    cities.addActionListener(this);
```

```
    states.add(new MenuItem("Himachal
Pradesh"));
    states.add(new
MenuItem("Rajasthan"));
    states.add(new MenuItem("West
Bengal"));
    states.addSeparator();
          // separates from north Indian
states from south Indian
    states.add(new MenuItem("Andhra
Pradesh"));
    states.add(new
MenuItem("Tamilnadu"));
    states.add(new
MenuItem("Karnataka"));

    cities.add(new MenuItem("Delhi"));
    cities.add(new MenuItem("Jaipur"));
    cities.add(new MenuItem("Kolkata"));
    cities.addSeparator();
          // separates north Indian cities
from south Indian
    cities.add(new
MenuItem("Hyderabad"));
    cities.add(new MenuItem("Chennai"));
    cities.add(new
MenuItem("Bengaluru"));

    add(ta, "Center");
    setTitle("Simple Menu Program");
          // frame creation methods
    setSize(300, 300);
    setVisible(true);
  }
  public void
actionPerformed(ActionEvent e)
  {
    String str = e.getActionCommand();
          // know the menu item selected
by the user
    ta.setText("You selected " + str);
  }
  public static void main(String args[])
  {
    new SimpleMenuExample();
  }
}
```

Program 7 – Focus Event with TextField Example

```
import java.awt.*;
import java.awt.event.*;
class FocusEv extends Frame
implements FocusListener
{
        TextField tf1,tf2;
        FocusEv()
        {
                tf1=new TextField (20);
      tf2=new TextField (20);
                setLayout(new
FlowLayout());
                add(tf1);
                add(tf2);

        tf1.addFocusListener(this);
                setSize(500,500);
                setVisible(true);
        }
        public static void main(String[]
args)
        {
                new FocusEv();
        }
        public void focusGained
(FocusEvent e)
        {
                if(e.getSource()==tf1)
                {
                        tf2.setText(" ");

        tf1.setText("@hotmail.com");
                }
        }
        public void focusLost
(FocusEvent e)
        {
                if(e.getSource()==tf1)
                {

        if(tf1.getText().length()<20)
                        {

        tf2.setText("Invalid");
                        }
                        else
                        {
```

```
        tf2.setText("Valid");
                        }
                }
        }
}
```

Program 8 – Item Event with CheckBoxGroup Example

```
import java.awt.*;
import java.awt.event.*;
class Greeting extends Frame
{
   CheckboxGroup cg;
   Checkbox morn,anoon,eve,nyt;
   String greet="";
   Font f;
   public Greeting()
   {
  setLayout(new FlowLayout());
setSize(300,300);
setTitle("Greetings  !!!!");
      cg = new CheckboxGroup();
      morn = new Checkbox("Morning",
cg, false);
 anoon = new Checkbox("Afternoon", cg,
false);
   eve = new Checkbox("Evening", cg,
false);
   nyt = new Checkbox("Night", cg,
false);

       add(morn);
       add(anoon);
       add(eve);
       add(nyt);

       addWindowListener (new
WindowAdapter()
       {
          public void
windowClosing(WindowEvent we)
          {
             setVisible(false);
             System.exit(0);
          }
       });
   }
public void
itemStateChanged(ItemEvent ie)
  {
f=new Font("Bookman Old
Style",Font.BOLD+Font.ITALIC,20);
CheckBox  sel=sel.getCurrent();
   if(sel.equals(morn))
          greet="Good Morning to you
Sir!!!!";
   else if(sel.equals(anoon))
            greet="Good Afternoon to you
Sir!!!!";
   else if(sel.equals(eve))
          greet="Good Evening to you
Sir!!!!";
else
greet="Good Night to you Sir!!!!"
   repaint();
  }
  public void paint(Graphics g)
  {
    g.setFont(f);
    g.drawString(greet,100,100);
}
   public static void main(String args[])
   {
       Greeting gt = new Greeting();

      gt.show();
   }
}
```

Program 9 – Item Event with Choice Example

```
import java.applet.*;
import java.awt.*;
import java.awt.event.*;
/*
<applet code="GamesChoice"
width=380 height=150>
</applet>
*/

public class GamesChoice extends
Applet  implements ItemListener
 {
  Choice c1;
  Color color;
String sel,msg="";
Font f;
```

```
  public void init()
{
    c1 = new Choice();
    c1.addItem("Hockey");
    c1.addItem("Tennis");
    c1.addItem("Football");
    c1.addItemListener(this);
    add(c1);

  }

public void
itemStateChanged(ItemEvent ie)
  {
 f=new Font("Bookman Old
Style",Font.BOLD+Font.ITALIC,20);
  sel=c1.getSelectedItem();
    if(sel.equals("Hockey"))
            msg="You are a Hockey
Player";
    else if(sel.equals("Tennis"))
            msg="You are a Tennis
Player";
    else
             msg="You are a Football
Player";
    repaint();
  }
  public void paint(Graphics g)
  {
     g.setFont(f);
      g.drawString(msg,100,100);
  }
}
```

Program 10 – Mouse Event Example

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*
<applet code="MouseEvents" width=300
height=100>
</applet>
*/
public class MouseEvents extends
Applet implements MouseListener,
MouseMotionListener
{
    String msg = "";
    int mouseX = 0, mouseY = 0; //
coordinates of mouse
    public void init()
    {
        addMouseListener(this);
        addMouseMotionListener(this);
    }
    // Handle mouse clicked.
    public void
mouseClicked(MouseEvent me)
    {
        // save coordinates
        mouseX = me.getX();
        mouseY = me.getY();
        msg = "Mouse clicked.";
        repaint();
    }
    // Handle mouse entered.
    public void
mouseEntered(MouseEvent me)
    {
        // save coordinates
        mouseX = 0;
        mouseY = 10;
        msg = "Mouse entered.";
        repaint();
    }
    // Handle mouse exited.
    public void mouseExited(MouseEvent
me)
    {
        // save coordinates
        mouseX = 0;
        mouseY = 10;
        msg = "Mouse exited.";
        repaint();
    }
    // Handle button pressed.
    public void
mousePressed(MouseEvent me)
    {
        // save coordinates
        mouseX = me.getX();
        mouseY = me.getY();
        msg = "Down";
        repaint();
    }
    // Handle button released.
    public void
mouseReleased(MouseEvent me)
```

```
    {
        // save coordinates
        mouseX = me.getX();
        mouseY = me.getY();
        msg = "Up";
        repaint();
    }
    // Handle mouse dragged.
    public void
mouseDragged(MouseEvent me)
    {
        // save coordinates
        mouseX = me.getX();
        mouseY = me.getY();
        msg = "Mouse Dragged";
        showStatus("Dragging mouse at "
+ mouseX + ", " + mouseY);
        repaint();
    }
    // Handle mouse moved.
    public void mouseMoved(MouseEvent
me)
    {
        // show status
        mouseX = me.getX();
        mouseY = me.getY();
        msg="Mouse Moved";
        showStatus("Moving mouse at " +
me.getX() + ", " + me.getY());
        repaint();
    }
    // Display msg in applet window at
current X,Y location.
    public void paint(Graphics g)
    {
        g.drawString(msg, mouseX,
mouseY);
    }
}
```

Program 11 – Key Event Example

```
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
/*<applet code="keyTest" width =400
height=300>
</applet>
*/
public class keyTest extends Applet
implements KeyListener
{
public void init()
{
Label lab = new Label ("Enter
Characters :");
add (lab);
TextField tf = new TextField (20);
add (tf);
tf.addKeyListener(this);
}
public void keyPressed(KeyEvent e)
{
showStatus("key Down");
}
public void keyReleased(KeyEvent e)
{
showStatus("key Up");
}
public void keyTyped(KeyEvent e)
{
showStatus(" Recently typed characters
are : " + e.getKeyChar());
}
}
```