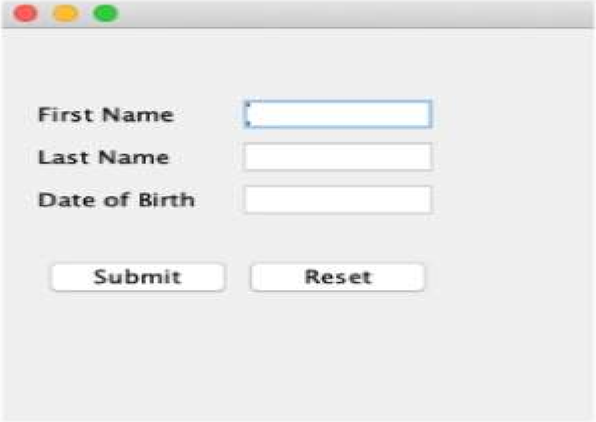
Java Awt Example

```
import java.awt.*;
public class AwtApp extends Frame {  // extending the frame package to use its components
AwtApp(){  // calling the awt app
Label firstName = new Label("First Name");  // creating label for first name
firstName.setBounds(20, 50, 80, 20);  // setting up the co-ordinates for the label

Label lastName = new Label("Last Name");  // creating label for lastname
lastName.setBounds(20, 80, 80, 20);  // creating the co-ordinates for the label

Label dob = new Label("Date of Birth");  // label for date of birth
dob.setBounds(20, 110, 80, 20);  // co-ordiantes of dob

TextField firstNameTF = new TextField();  // creating text area for first name
firstNameTF.setBounds(120, 50, 100, 20);  // co-ordinates for firstname

TextField lastNameTF = new TextField();  // creating textarea for last name
lastNameTF.setBounds(120, 80, 100, 20);  // co-ordinates for lastname

TextField dobTF = new TextField();  // textarea for dob
dobTF.setBounds(120, 110, 100, 20);  // co-ordinates

Button sbmt = new Button("Submit");  // creating submit button
sbmt.setBounds(20, 160, 100, 30);  // co-ordinates of submit button

Button reset = new Button("Reset");  // creating reset button
reset.setBounds(120,160,100,30);  // co-ordinates of reset button

add(firstName);  // adding first name component on the window object
add(lastName);  //adding last name component on the window object
add(dob);  // adding dob component on the window object
add(firstNameTF);  // adding first name text area component on the window object
add(lastNameTF);  //adding last name textarea component on the window object
add(dobTF);  //adding dob textarea component on the window object
add(sbmt);  //adding submit button component on the window object
add(reset);  //adding reset button component on the window object

setSize(300,300);  // setting up the window co-ordinates
setLayout(null);  // setting the layout to null
setVisible(true);  // setting up window’s visiblity
}
public static void main(String[] args) {
// TODO Auto-generated method stub
AwtApp awt = new AwtApp();  // initializing awt object

}
}
```

Output



Java Swing Example

```
import javax.swing.*;
public class SwingApp {
SwingApp(){  // initialzing the swing app
JFrame f = new JFrame();  // initializing the new JFrame

JLabel firstName = new JLabel("First Name");  // creating object of firstname
firstName.setBounds(20, 50, 80, 20);   // setting up the co-ordinates

JLabel lastName = new JLabel("Last Name");  //creating object of lastname
lastName.setBounds(20, 80, 80, 20);   // setting up the co-ordinates

JLabel dob = new JLabel("Date of Birth");  // creating object for date of birth
dob.setBounds(20, 110, 80, 20);   // setting up the co-ordinates

JTextField firstNameTF = new JTextField();  // creating firstname text area
firstNameTF.setBounds(120, 50, 100, 20);   // setting up the co-ordinates

JTextField lastNameTF = new JTextField();  // creating text area for last name
lastNameTF.setBounds(120, 80, 100, 20);   // setting up the co-ordinates

JTextField dobTF = new JTextField();  // creating text area for date of birth
dobTF.setBounds(120, 110, 100, 20);   // setting up the co-ordinates

JButton sbmt = new JButton("Submit");  // creating submit button
sbmt.setBounds(20, 160, 100, 30);   // setting up the co-ordinates

JButton reset = new JButton("Reset");  // creating reset button
reset.setBounds(120,160,100,30);   // setting up the co-ordinates

f.add(firstName);  // adding first name component to the window
f.add(lastName);  // adding last name component to the window
f.add(dob);  // adding date of birth component onto the window
f.add(firstNameTF);  // adding text area of firstname
f.add(lastNameTF);  // adding text area for lastname
f.add(dobTF);  // adding text area for date of birth
f.add(sbmt);  // adding submit button component onto the window
f.add(reset);  // adding reset button

f.setSize(300,300);   // setting up the co-ordinates for window size
f.setLayout(null);  // setting up a layout
f.setVisible(true);  // setting up the visibility of a window object
}

public static void main(String[] args) {
// TODO Auto-generated method stub
SwingApp s = new SwingApp();  // calling the swing app method
}
}
```

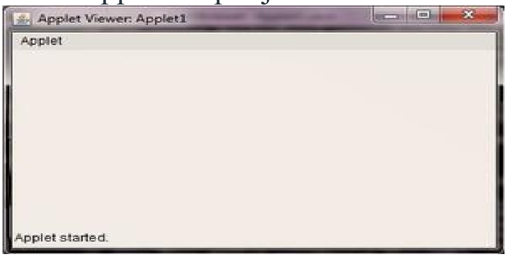
Applet Program

```
// This is a Simple Java Applet program using appletviewer

import java.applet.*;
import java.awt.*;

/*
<applet code="AppletExp1" width=600 height=300>
</applet>
*/

public class AppletExp1 extends Applet {
        public void init()
        {
                System.out.println("Initializing an applet");
        }

        public void start()
        {
                System.out.println("Starting an applet");
        }
        public void stop()
        {
                System.out.println("Stopping an applet");
        }
        public void destroy()
        {
                System.out.println("Destroying an applet");
        }
}
```

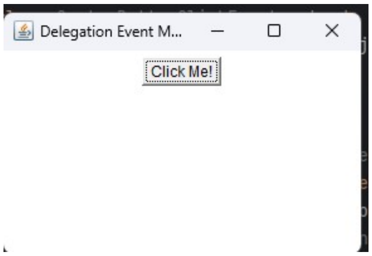
By using Appletviewer, type the following command at the command prompt

```
appletviewer AppleteExpt1.java
```

Output:

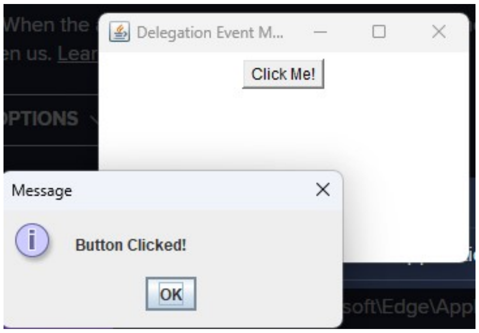


Applet started.



Java Program to Implement the Event Deligation Model

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class DelegationEventModelDemo {
   public static void main(String[] args) {
      // Step 1: Creating Custom Events
      class CustomButtonClickEvent extends java.util.EventObject {
         public CustomButtonClickEvent(Object source) {
            super(source);
         }
      }
      // Step 2: Implementing Event Listeners
      class ActionButtonClickListener implements ActionListener {
         public void actionPerformed(ActionEvent e) {
            // Show a message dialog when the button is clicked
            JOptionPane.showMessageDialog(null, "Button Clicked!");
         }
      }
      // Step 3: Registering and Handling Events
      Frame frame = new Frame("Delegation Event Model Demo");
      frame.setLayout(new FlowLayout());
      Button button = new Button("Click Me!");
      frame.add(button);
      ActionButtonClickListener actionButtonClickListener = new ActionButtonClickListener();
      button.addActionListener(actionButtonClickListener);
      frame.setSize(300, 200);
      frame.setVisible(true);
      frame.addWindowListener(new WindowAdapter() {
         public void windowClosing(WindowEvent e) {
            System.exit(0);
         }
      });
   }
}
```



Button Clicked!

```
//Basic Calculator using action listener
// basic calculator
import java.awt.*;
import java.awt.event.*;
class Main extends WindowAdapter implements ActionListener{
    Frame f;
    Label l1;
    Button b1,b2,b3,b4,b5,b6,b7,b8,b9,b0;
    Button badd,bsub,bmult,bdiv,bmod,bcalc,bclr,bpts,bneg,bback;
    double xd;
    double num1,num2,check;

    Main(){
        f= new Frame("MY CALCULATOR");
// INSTANTIATING COMPONENETS
        // Step 1: Designing the GUI Interface
        l1=new Label();
        l1.setBackground(Color.LIGHT_GRAY);
        l1.setBounds(50,50,260,60);
        l1.setVisible(true);

        b1=new Button("1");
        b1.setBounds(50,340,50,50);
        b2=new Button("2");
        b2.setBounds(120,340,50,50);
        b3=new Button("3");
        b3.setBounds(190,340,50,50);
        b4=new Button("4");
        b4.setBounds(50,270,50,50);
        b5=new Button("5");
        b5.setBounds(120,270,50,50);
        b6=new Button("6");
        b6.setBounds(190,270,50,50);
        b7=new Button("7");
        b7.setBounds(50,200,50,50);
        b8=new Button("8");
        b8.setBounds(120,200,50,50);
        b9=new Button("9");
        b9.setBounds(190,200,50,50);
        b0=new Button("0");
        b0.setBounds(120,410,50,50);
        bneg=new Button("+/-");
        bneg.setBounds(50,410,50,50);
        bpts=new Button(".");
        bpts.setBounds(190,410,50,50);
        bback=new Button("back");
        bback.setBounds(120,130,50,50);
        badd=new Button("+");
        badd.setBounds(260,340,50,50);
        bsub=new Button("-");
        bsub.setBounds(260,270,50,50);
        bmult=new Button("*");
        bmult.setBounds(260,200,50,50);
        bdiv=new Button("/");
        bdiv.setBounds(260,130,50,50);
        bmod=new Button("%");
        bmod.setBounds(190,130,50,50);
        bcalc=new Button("=");
        bcalc.setBounds(245,410,65,50);
        bclr=new Button("CE");
        bclr.setBounds(50,130,65,50);

        // Step 2: Implementing Event Listeners
        // Create and register event listeners for each button

        b1.addActionListener(this);
        b2.addActionListener(this);
        b3.addActionListener(this);
        b4.addActionListener(this);
        b5.addActionListener(this);
        b6.addActionListener(this);
        b7.addActionListener(this);
        b8.addActionListener(this);
        b9.addActionListener(this);
        b0.addActionListener(this);

        bpts.addActionListener(this);
        bneg.addActionListener(this);
        bback.addActionListener(this);

        badd.addActionListener(this);
        bsub.addActionListener(this);
        bmult.addActionListener(this);
        bdiv.addActionListener(this);
        bmod.addActionListener(this);
        bcalc.addActionListener(this);
        bclr.addActionListener(this);
        f.addWindowListener(this);
//ADDING TO FRAME
        f.add(l1);
        f.add(b1); f.add(b2); f.add(b3); f.add(b4); f.add(b5);f.add(b6); f.add(b7);
f.add(b8);f.add(b9);f.add(b0);
        f.add(badd); f.add(bsub); f.add(bmod); f.add(bmult); f.add(bdiv); f.add(bmod);f.add(bcalc);
        f.add(bclr); f.add(bpts);f.add(bneg); f.add(bback);
```

```
    f.setSize(360,500);
    f.setLayout(null);
    f.setVisible(true);
}
//FOR CLOSING THE WINDOW
public void windowClosing(WindowEvent e) {
    f.dispose();
}

// Step 3: Event Handling in the Application
// Event handling is done through the actionPerformed method of the event listener
// When a button is clicked, its event listener will execute the corresponding operation

public void actionPerformed(ActionEvent e){
    String z,zt;
    //NUMBER BUTTON
    if(e.getSource()==b1){
        zt=l1.getText();
        z=zt+"1";
        l1.setText(z);
    }
    if(e.getSource()==b2){
        zt=l1.getText();
        z=zt+"2";
        l1.setText(z);
    }
    if(e.getSource()==b3){
        zt=l1.getText();
        z=zt+"3";
        l1.setText(z);
    }
    if(e.getSource()==b4){
        zt=l1.getText();
        z=zt+"4";
        l1.setText(z);
    }
    if(e.getSource()==b5){
        zt=l1.getText();
        z=zt+"5";
        l1.setText(z);
    }
    if(e.getSource()==b6){
        zt=l1.getText();
        z=zt+"6";
        l1.setText(z);
    }
    if(e.getSource()==b7){
        zt=l1.getText();
```

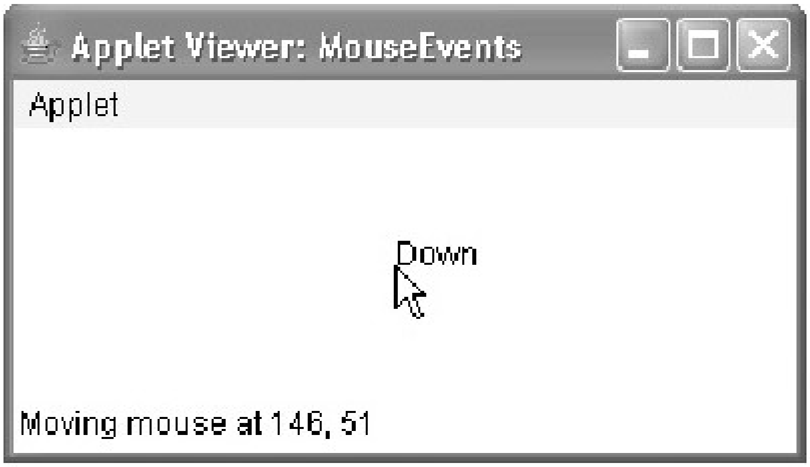
```
   z=zt+"7";
   l1.setText(z);
}
if(e.getSource()==b8){
   zt=l1.getText();
   z=zt+"8";
   l1.setText(z);
}
if(e.getSource()==b9){
   zt=l1.getText();
   z=zt+"9";
   l1.setText(z);
}
if(e.getSource()==b0){
   zt=l1.getText();
   z=zt+"0";
   l1.setText(z);
}

if(e.getSource()==bpts){  //ADD DECIMAL PTS
   zt=l1.getText();
   z=zt+".";
   l1.setText(z);
}
if(e.getSource()==bneg){ //FOR NEGATIVE
   zt=l1.getText();
   z="-"+zt;
   l1.setText(z);
}

if(e.getSource()==bback){  // FOR  BACKSPACE
   zt=l1.getText();
   try{
      z=zt.substring(0, zt.length()-1);
   }catch(StringIndexOutOfBoundsException f){return;}
   l1.setText(z);
}
//AIRTHMETIC BUTTON
if(e.getSource()==badd){                    //FOR ADDITION
   try{
      num1=Double.parseDouble(l1.getText());
   }catch(NumberFormatException f){
      l1.setText("Invalid Format");
      return;
   }
   z="";
   l1.setText(z);
   check=1;
```

```
}
if(e.getSource()==bsub){                    //FOR SUBTRACTION
   try{
      num1=Double.parseDouble(l1.getText());
   }catch(NumberFormatException f){
      l1.setText("Invalid Format");
      return;
   }
   z="";
   l1.setText(z);
   check=2;
}
if(e.getSource()==bmult){                   //FOR MULTIPLICATION
   try{
      num1=Double.parseDouble(l1.getText());
   }catch(NumberFormatException f){
      l1.setText("Invalid Format");
      return;
   }
   z="";
   l1.setText(z);
   check=3;
}
if(e.getSource()==bdiv){                    //FOR DIVISION
   try{
      num1=Double.parseDouble(l1.getText());
   }catch(NumberFormatException f){
      l1.setText("Invalid Format");
      return;
   }
   z="";
   l1.setText(z);
   check=4;
}
if(e.getSource()==bmod){                    //FOR MOD/REMAINDER
   try{
      num1=Double.parseDouble(l1.getText());
   }catch(NumberFormatException f){
      l1.setText("Invalid Format");
      return;
   }
   z="";
   l1.setText(z);
   check=5;
}
//RESULT BUTTON
if(e.getSource()==bcalc){
   try{
```

```
                num2=Double.parseDouble(l1.getText());
            }catch(Exception f){
                l1.setText("ENTER NUMBER FIRST ");
                return;
            }
            if(check==1)
                xd =num1+num2;
            if(check==2)
                xd =num1-num2;
            if(check==3)
                xd =num1*num2;
            if(check==4)
                xd =num1/num2;
            if(check==5)
                xd =num1%num2;
            l1.setText(String.valueOf(xd));
        }
        //FOR CLEARING THE LABEL and Memory
        if(e.getSource()==bclr){
            num1=0;
            num2=0;
            check=0;
            xd=0;
            z="";
            l1.setText(z);
        }

    }
    //MAIN METHOD where objects of Main is instantaiated
    public static void main(String args[]){
        new Main();
    }
}
```

Event Handler using Applet

```
// Demonstrate the mouse event handlers.
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 = 0;
mouseY = 10;
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 = "*";
showStatus("Dragging mouse at " + mouseX + ", " + mouseY);
repaint();
}
// Handle mouse moved.
public void mouseMoved(MouseEvent me) {
// show status
showStatus("Moving mouse at " + me.getX() + ", " + me.getY());
}
// Display msg in applet window at current X,Y location.
public void paint(Graphics g) {
g.drawString(msg, mouseX, mouseY);
}
}
```

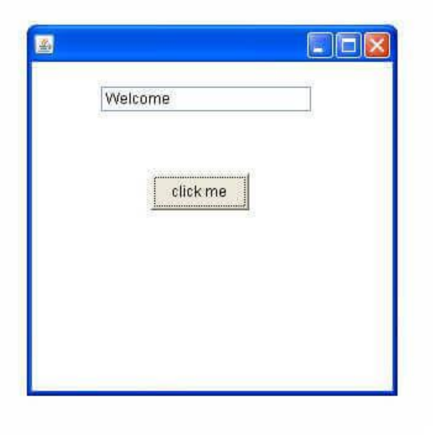


```
Down
```

```
Moving mouse at 146, 51
```

Java event handling by implementing ActionListener

```
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();
}
}
```



Welcome

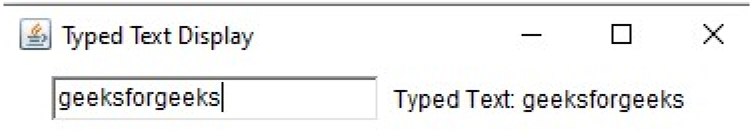
Java event handling by outer class

```
import java.awt.*;
import java.awt.event.*;
class AEvent2 extends Frame{
TextField tf;
AEvent2(){
//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
Outer o=new Outer(this);
b.addActionListener(o);//passing outer class instance
//add components and set size, layout and visibility
add(b);add(tf);
setSize(300,300);
setLayout(null);
setVisible(true);
}
public static void main(String args[]){
new AEvent2();
}
}


import java.awt.event.*;
class Outer implements ActionListener{
AEvent2 obj;
Outer(AEvent2 obj){
this.obj=obj;
}
public void actionPerformed(ActionEvent e){
obj.tf.setText("welcome");
}
}
```

Java event handling by anonymous class

```
import java.awt.*;
import java.awt.event.*;
class AEvent3 extends Frame{
TextField tf;
AEvent3(){
tf=new TextField();
tf.setBounds(60,50,170,20);
Button b=new Button("click me");
b.setBounds(50,120,80,30);

b.addActionListener(new ActionListener(){
public void actionPerformed(){
tf.setText("hello");
}
});
add(b);add(tf);
setSize(300,300);
setLayout(null);
setVisible(true);
}
public static void main(String args[]){
new AEvent3();
}
}
```

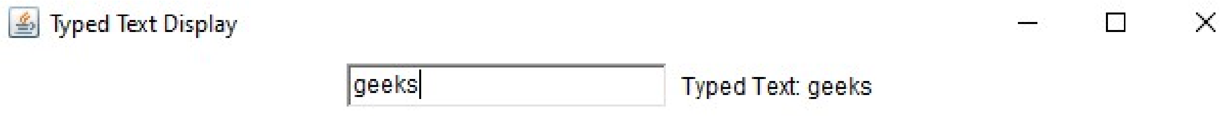
```
// Java program to demonstrate textfield and display typed text using KeyListener
import java.awt.*;
import java.awt.event.*;

public class KeyListenerExample extends Frame implements KeyListener {

    private TextField textField;
    private Label displayLabel;

    // Constructor
    public KeyListenerExample() {
        // Set frame properties
        setTitle("Typed Text Display");
        setSize(400, 200);
        setLayout(new FlowLayout());

        // Create and add a TextField for text input
        textField = new TextField(20);
        textField.addKeyListener(this);
        add(textField);

        // Create and add a Label to display typed text
        displayLabel = new Label("Typed Text: ");
        add(displayLabel);

        // Ensure the frame can receive key events
        setFocusable(true);
        setFocusTraversalKeysEnabled(false);

        // Make the frame visible
        setVisible(true);
    }

    // Implement the keyPressed method
    @Override
    public void keyPressed(KeyEvent e) {
        // You can add custom logic here if needed
    }

    // Implement the keyReleased method
    @Override
    public void keyReleased(KeyEvent e) {
        // You can add custom logic here if needed
    }

    // Implement the keyTyped method
    @Override
    public void keyTyped(KeyEvent e) {
        char keyChar = e.getKeyChar();
        displayLabel.setText("Typed Text: " + textField.getText() + keyChar);
    }

    public static void main(String[] args) {
        new KeyListenerExample();
    }
}
```



geeksforgeeks

Typed Text: geeksforgeeks

```
//Java program to demonstrate keyPressed,  keyReleased and keyTyped method
import java.awt.*;
import java.awt.event.*;

public class KeyListenerExample extends Frame implements KeyListener {

    private TextField textField;
    private Label displayLabel;

    // Constructor
    public KeyListenerExample() {
        // Set frame properties
        setTitle("Typed Text Display");
        setSize(400, 200);
        setLayout(new FlowLayout());

        // Create and add a TextField for text input
        textField = new TextField(20);
        textField.addKeyListener(this);
        add(textField);

        // Create and add a Label to display typed text
        displayLabel = new Label("Typed Text: ");
        add(displayLabel);

        // Ensure the frame can receive key events
        setFocusable(true);
        setFocusTraversalKeysEnabled(false);

        // Make the frame visible
        setVisible(true);
    }

    // Implement the keyPressed method
    @Override
    public void keyPressed(KeyEvent e) {
        int keyCode = e.getKeyCode();
        System.out.println("Key Pressed: " + KeyEvent.getKeyText(keyCode));
    }

    // Implement the keyReleased method
    @Override
    public void keyReleased(KeyEvent e) {
        int keyCode = e.getKeyCode();
        System.out.println("Key Released: " + KeyEvent.getKeyText(keyCode));
    }

    // Implement the keyTyped method
    @Override
    public void keyTyped(KeyEvent e) {
        char keyChar = e.getKeyChar();
        System.out.println("Key Typed: " + keyChar);
        displayLabel.setText("Typed Text: " + textField.getText() + keyChar);
    }

    public static void main(String[] args) {
        new KeyListenerExample();
    }
}
```

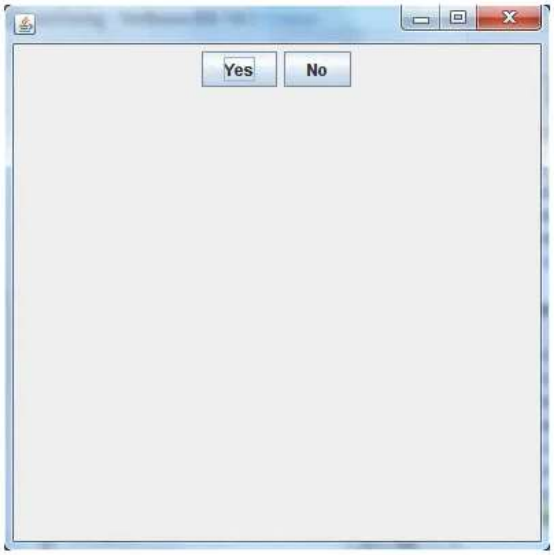


geeks

Typed Text: geeks

Swing JButton

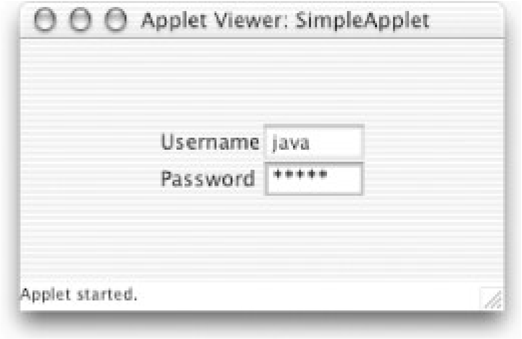
```
import javax.swing.*;
import java.awt.event.*;
import java.awt.*;

public class testswing extends JFrame
{

  testswing()
  {
    JButton bt1 = new JButton("Yes");            //Creating a Yes Button.
    JButton bt2 = new JButton("No");             //Creating a No Button.
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE)     //setting close operation.
    setLayout(new FlowLayout());          //setting layout using FlowLayout object
    setSize(400, 400);                    //setting size of Jframe
    add(bt1);             //adding Yes button to frame.
    add(bt2);             //adding No button to frame.
    setVisible(true);
  }

  public static void main(String[] args)
  {
    new testswing();
  }
}
```



JApplet program

```
import javax.swing.*;
import javax.swing.border.*;
import java.awt.*;

public class SimpleApplet extends JApplet {
  public void init( ) {
    JPanel p = new JPanel( );
    p.setLayout(new GridLayout(2, 2, 2, 2));
    p.add(new JLabel("Username"));
    p.add(new JTextField( ));
    p.add(new JLabel("Password"));
    p.add(new JPasswordField( ));
    Container content = getContentPane( );
    content.setLayout(new GridBagLayout( )); // Used to center the panel
    content.add(p);
  }
}
```

JApplet Output

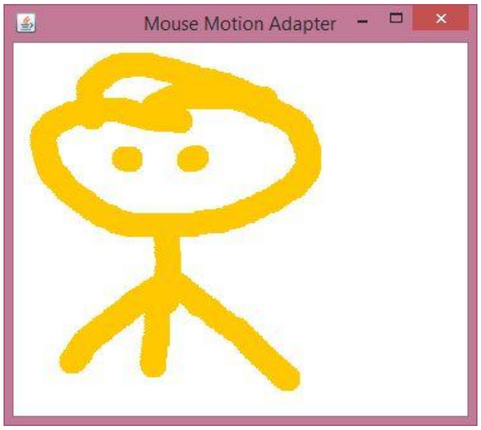


Username java
Password *****

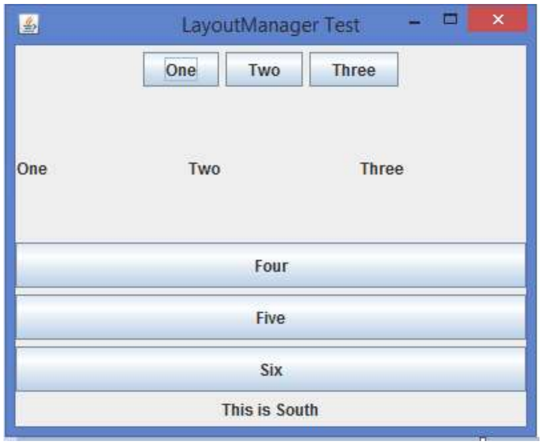
Applet started.

Adaptor class for Mouse motion

```
// importing the necessary libraries
import java.awt.*;
import java.awt.event.*;
// class which inherits the MouseMotionAdapter class
public class MouseMotionAdapterExample extends MouseMotionAdapter {
// object of Frame class
    Frame f;
// class constructor
    MouseMotionAdapterExample() {
// creating the frame with the title
        f = new Frame ("Mouse Motion Adapter");
// adding MouseMotionListener to the Frame
        f.addMouseMotionListener (this);
 // setting the size, layout and visibility of the frame
        f.setSize (300, 300);
        f.setLayout (null);
        f.setVisible (true);
    }
// overriding the mouseDragged() method
public void mouseDragged (MouseEvent e) {
// creating the Graphics object and fetching them from the Frame object using getGraphics() method
    Graphics g = f.getGraphics();
// setting the color of graphics object
    g.setColor (Color.ORANGE);
// setting the shape of graphics object
    g.fillOval (e.getX(), e.getY(), 20, 20);
}
public static void main(String[] args) {
    new MouseMotionAdapterExample();
}
}
```



Layout Manager: Border Layout



```
import java.awt.*;
import javax.swing.*;
public class LayoutManagerTest extends JFrame {
  JPanel flowLayoutPanel1, flowLayoutPanel2, gridLayoutPanel1, gridLayoutPanel2,
gridLayoutPanel3;
  JButton one, two, three, four, five, six;
  JLabel bottom, lbl1, lbl2, lbl3;
  public LayoutManagerTest() {
    setTitle("LayoutManager Test");
    setLayout(new BorderLayout()); // Set BorderLayout for JFrame
    flowLayoutPanel1 = new JPanel();
    one = new JButton("One");
    two = new JButton("Two");
    three = new JButton("Three");
    flowLayoutPanel1.setLayout(new FlowLayout(FlowLayout.CENTER)); // Set FlowLayout
Manager
    flowLayoutPanel1.add(one);
    flowLayoutPanel1.add(two);
    flowLayoutPanel1.add(three);
    flowLayoutPanel2 = new JPanel();
    bottom = new JLabel("This is South");
    flowLayoutPanel2.setLayout (new FlowLayout(FlowLayout.CENTER));
// Set FlowLayout Manager
    flowLayoutPanel2.add(bottom);
    gridLayoutPanel1 = new JPanel();
    gridLayoutPanel2 = new JPanel();
    gridLayoutPanel3 = new JPanel();
    lbl1 = new JLabel("One");
    lbl2 = new JLabel("Two");
    lbl3 = new JLabel("Three");
    four = new JButton("Four");
    five = new JButton("Five");
    six = new JButton("Six");
    gridLayoutPanel2.setLayout(new GridLayout(1, 3, 5, 5)); // Set GridLayout Manager
    gridLayoutPanel2.add(lbl1);
    gridLayoutPanel2.add(lbl2);
    gridLayoutPanel2.add(lbl3);
    gridLayoutPanel3.setLayout(new GridLayout(3, 1, 5, 5)); // Set GridLayout Manager
    gridLayoutPanel3.add(four);
    gridLayoutPanel3.add(five);
    gridLayoutPanel3.add(six);
    gridLayoutPanel1.setLayout(new GridLayout(2, 1)); // Set GridLayout Manager
    gridLayoutPanel1.add(gridLayoutPanel2);
    gridLayoutPanel1.add(gridLayoutPanel3);
    add(flowLayoutPanel1, BorderLayout.NORTH);
    add(flowLayoutPanel2, BorderLayout.SOUTH);
    add(gridLayoutPanel1, BorderLayout.CENTER);
    setSize(400, 325);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setLocationRelativeTo(null);
    setVisible(true);
  }
  public static void main(String args[]) {
    new LayoutManagerTest();
  }
}
```

```
//Inner class
class Listener {
class oneBtnListener implements ActionListener {
   @Override
  public void actionPerformed(ActionEvent ev) {
      if(ev.getActionCommand() == "1")
      {
          JButton btn = (JButton)ev.getSource();
          btn.setText("11");
      }
  }
}
}

class calculatorGUI extends javax.swing.JFrame {
public calculatorGUI() {
   initComponents();
}
private void initComponents() {
oneBtn = new javax.swing.JButton();
oneBtnListener btnListener = new Listener().new oneBtnListener();
oneBtn.setText("1");
oneBtn.setBounds(100,100,100,25);
oneBtn.addActionListener(btnListener);
add(oneBtn);
setLayout(null);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setSize(400,400);
}

private javax.swing.JButton oneBtn;
public JButton getOneBtn() {
   return oneBtn;
}

public void setOneBtn(String name) {
   oneBtn.setText(name);
}
}
```

Anonymous Inner Class

```
package simple.button;
public class Main {
        public static void main(String[] args) {
                CustomJFrame myFrame = new CustomJFrame();
        }
}

package simple.button;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JPanel;

public class CustomJFrame extends JFrame{
        private JPanel contentPane;
//This is an invisible layer on our JFrame, to which we attach our button and any other
components we want shown
        private JButton myButton;
//This is the button we will attach to the contentPane

        public CustomJFrame() {
          this.setSize(200,70);       //This sets the dimensions of the JFrame
          contentPane = new JPanel();  //this creates a contentPane for holding our button
          myButton = new JButton("My Button");
          myButton.addActionListener(
         //here we use an anonymous inner class for the listener
                  new ActionListener() {
                     public void actionPerformed(ActionEvent e) {
                       System.out.println("My button was clicked!");
                     }
                  }
                );

          contentPane.add(myButton);
//This attaches the button to the contentPane.

          this.add(contentPane);
//This, in turn, attaches the contentPane to the JFrame.
          this.setVisible(true);     //This makes our JFrame visible.
        }
}
```