Dr. S & SS Ghandhy College Of Engineering & Technology, Surat

Lab Manual

Advance Java Programming (3360701)

Information Technology

Index

Sr. No.	Practical Name	Page No	Date	Sign
1	Develop an applet that draws a circle. The dimension of the applet should be 500 x 300 pixels. The circle should be centered in the applet and have a radius of 100 pixels. Display your name centered in a circle.(using drawOval() method)			
2	Draw ten red circles in a vertical column in the center of the applet.			
3	Built an applet that displays a horizontal rectangle in its center. Let the rectangle fill with color from left to right.			
4	Develop an applet that display the position of the mouse at the upper left corner of the applet when it is dragged or moved. Draw a 10x10 pixel rectangle filed with black at the current mouse position.			
5	Develop an applet that contains one button. Initialize the label on the button to “start”, when the user presses the button, which changes the label between these two values each time the button is pressed.			
6	Develop an applet that uses the mouse listener, which overrides only two methods which are mousePressed and mouseReleased.			
7	Develop a program that has only one button in the frame, clicking on the button cycles through the colors: red->green->blue and so on. One color changes per click.(use getBackGround() method to get the current color)			
8	Develop an program that contains three check boxes and 30 x 30 pixel canvas.The three checkboxes should be labeled “Red”, “Green”,”Blue”. The selection of the check boxes determine the color of the canvas. For example, if the user selects both “Red” and “Blue”, the canvas should be purple.			
9	Create an application that displays a frame with a menu bar. When a user selects any menu or menu item, display that selection on a text area in the center of the frame			
10	Develop a program that draws two sets of ever-decreasing rectangles one in outline form and one filled alternately in black and white.			
11	Develop a database application that uses any JDBC driver			
12	Develop a Graphical User Interface that performs the following SQL operations: a) Insert b) Delete c)Update.			
13	Develop a program to present a set of choice for user to select a product and display the price of product.			

14	Develop a simple servlet program which maintains a counter for the number of times it has been accessed since its loading, initialize the counter using deployment descriptor.			
15	Create a web form which processes servlet and demonstrates use of cookies and sessions.			
16	Develop a simple JSP program for user registration and then control will be transfer it into second page.			
17	Develop a simple JSP program for user login form with static and dynamic database			
18	Develop a JSP program to display the grade of a student by accepting the marks of five subjects.			

Dr S & SS Ghandhy College of Engineering & Technology, SURAT

Information Technology

Advance Java Programming

Course code: 3360701 **Credits: 6**

L: T: P: 3:0:4 **Total Marks:** 200

PU=Problem Understanding, P=Presentation/Demonstration

Criteria	Marks	Excellent (5)	Very Good(4)	Good (3)	fair (2)	Poor(1)
Problem Understanding (PU)	5	Able to define & explain concept properly and able to identify application of concept.	Able to identify all the problem/Task and able to apply all concept in problem/Task properly with very few help.	Able to identify almost all problem/Task and able to apply almost all concept in problem/Task with few exceptions.	Able to identify some problem/Task and able to apply some concept in problem/Task.	Able to identify very few problem/Task and able to apply very few concept in problem/Task.
Demonstration/ Presentation/Performance (D)	5	Able to run program correctly without any logic error and display appropriate output.	Follow all the procedure for experiment execution but display inappropriate output.	Follow almost all of the procedure for experiments, execution with few exceptions, few error or bug in output.	Able to construct statements but syntax error in program.	Unable to construct statement of program.

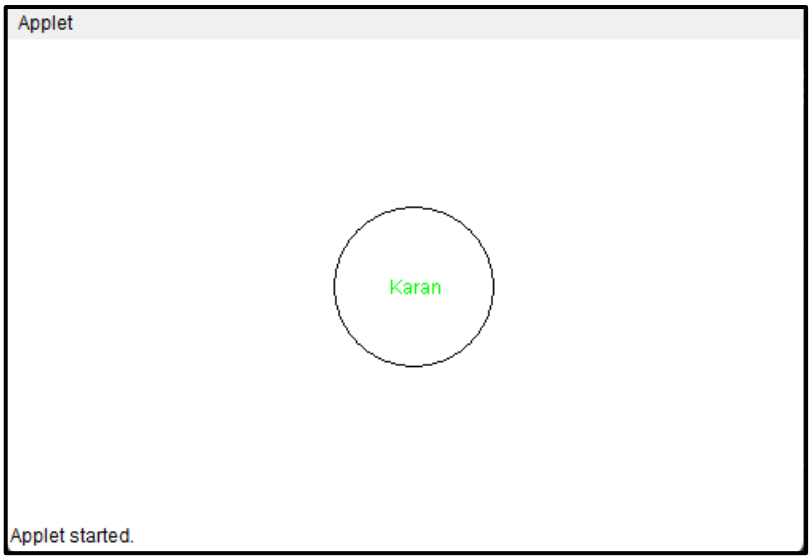
Practical 1

Aim : Develop an applet that draws a circle. The dimension of the applet should be 500 x 300 pixels. The circle should be centered in the applet and have a radius of 100 pixels. Display your name centered in a circle.(using drawOval() method).

Program :

```
package com.company;
import java.applet.*;
import java.awt.*; // Import Applet All Class Directory

/* Applet Width and Height is 500*300  */
public class Circle extends Applet {

   public void init()
   // Init() Method Invoke
   {
      setSize(500,300); // Width and Height of Applet
   }

   public void start()
   // Start() Method invoked
   {

   }

   public void paint(Graphics g)
   // Paint() Method invoked
   {
      g.drawOval(110+95,110-5,100,100); // Center Circle
      g.setColor(Color.GREEN); // Text Name
      g.drawString("Karan",250-10,150+10); // Text
   }
}
```

Output:


Applet
Karan
Applet started.

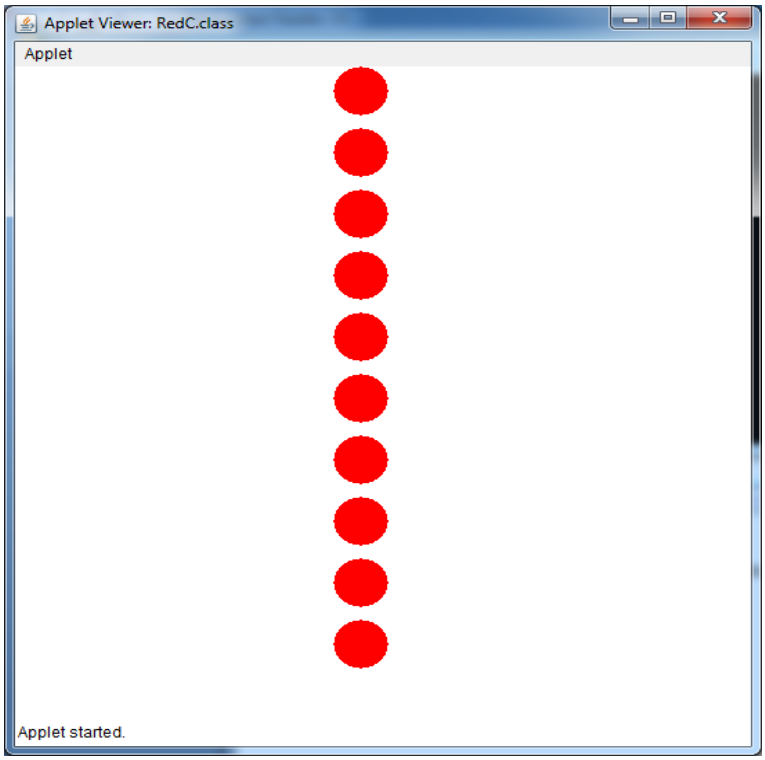
Practical 2

Aim :- Draw ten red circles in a vertical column in the center of the applet.

Program :

```
package com.company;

import java.applet.*;

import java.awt.*;

/* Width & Height Of Applet is 1000*1000 */

public class Circle10 extends Applet{

   public void init() // Init() Method Invoke

   {       setSize(1000,1000);    }

   public void start() // Start() Method Invoke

   {  }

   public void paint(Graphics g) // Paint() Method Invoke

   {

      int x=500;

      for(int i=1,y=0;i<11;i++,y+=50)

      {

         g.setColor(Color.RED);

         g.fillOval(x,y,40,40);

      }

   }

}
```

Output:


Applet Viewer: RedC.class
Applet
Applet started.

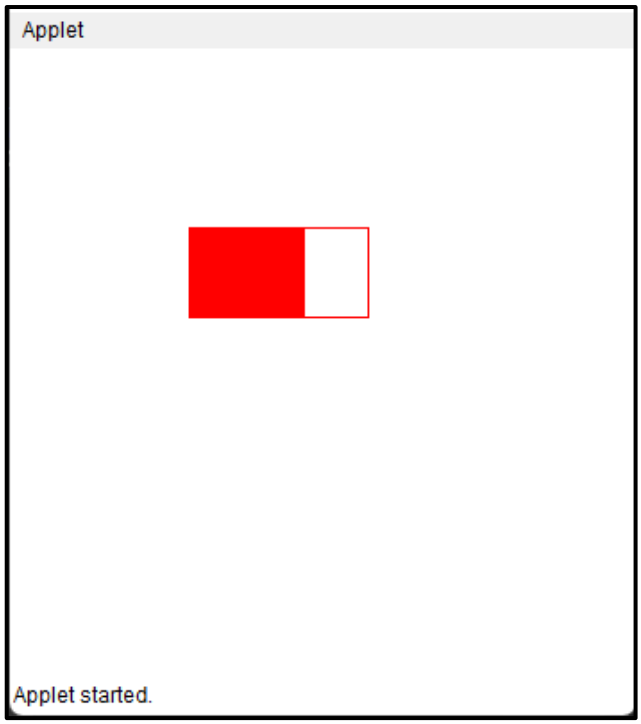
Practical 3

AIM :- Built an applet that displays a horizontal rectangle in its center. Let the rectangle fill with color from left to right.

Program :

```
package com.company;
import java.awt.*;
import java.applet.Applet;
/* Width & Height of Applet is 350*350 */
public class Rectangle extends Applet {

   public void init() // Init() Method Invoke
   {
      super.init();
      setSize(350,350);
   }

   public void paint(Graphics s) // Paint() Method Invoke
   {
      int x1=100,y1=100,y2=50;
      s.setColor(Color.green);
      s.drawRect(100, 100, 100, 50);

      for(x1=100;x1<300;x1=x1+5)
      {
         try
         {
            Thread.sleep(1000);
            s.fillRect(x1, y1, 5, y2);
         }
         catch(Exception e)
         {
            e.printStackTrace();
         }
      }
   }
}
```

Output:


Applet
Applet started.

Practical 4

Aim : Develop an applet that display the position of the mouse at the upper left corner of the applet when it is dragged or moved. Draw a 10×10 pixel rectangle filed with black at the current mouse position.

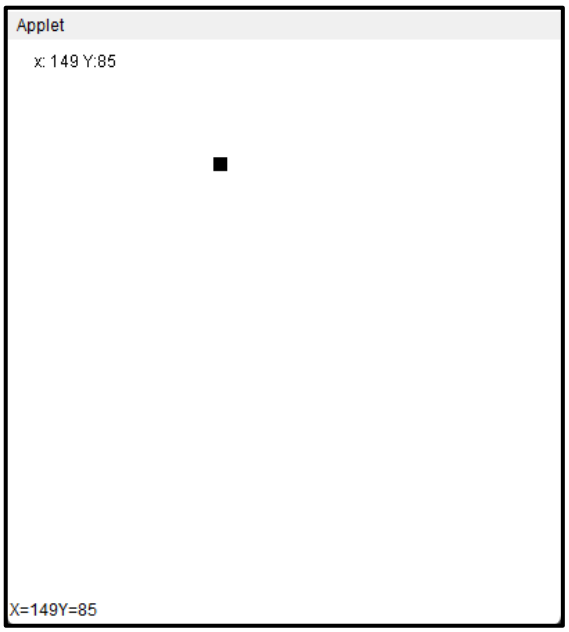
Program :

```
package com.company;
import java.applet.Applet;
import java.awt.*;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;
import java.awt.event.MouseMotionListener;
/* Width & Heigth of Applet is 400*400 */

public class Drag extends Applet implements MouseListener,MouseMotionListener  {

   String msg="";

   int x,y;
   public void init()
   // Init() Method Invoke
   {
      super.init();
      setSize(400,400);
      addMouseListener(this);
      addMouseMotionListener(this);
   }

   public void paint(Graphics s)
      // Paint() Method Invoke
   {

      showStatus(msg);
      s.fillRect(x, y, 10, 10);
      s.drawString("x: "+x+" Y:"+y, 20,20);
   }

   @Override
   public void mouseDragged(MouseEvent e) {
      // MouseDragged() Method Invoke
      x=e.getX();
      y=e.getY();
      msg="X="+x+"Y="+y;
```

```
    repaint();

  }

  @Override
  public void mouseMoved(MouseEvent e) {
    // MouseMoved() Method Invoke

    x=e.getX();
    y=e.getY();

    repaint();
    msg="X="+x+"Y="+y;

  }

  @Override
  public void mouseClicked(MouseEvent arg0) {
    // MouseClicked() Method Invoke

  }

  @Override
  public void mouseEntered(MouseEvent arg0) {
    // MouseEntered() Method Invoke

  }

  @Override
  public void mouseExited(MouseEvent arg0) {
    // MouseExited() Method Invoke

  }

  @Override
  public void mousePressed(MouseEvent arg0) {
    // MousePressed() Method Invoke

  }

  @Override
  public void mouseReleased(MouseEvent arg0) {
    // MouseReleased() Method Invoke
```

```
    }
}
```

Output:



Practical 5

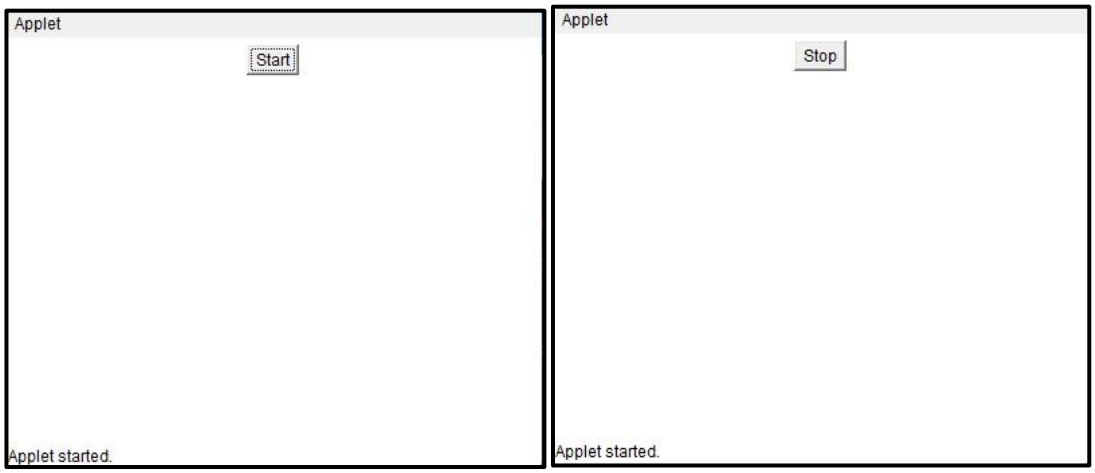
Aim :- Develop an applet that contains one button. Initialize the label on the button to “start”, when the user presses the button, which changes the label between these two values each time the button is pressed.

Program :

```
package com.company;

import java.applet.Applet;

import java.awt.*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class Practical5 extends Applet implements ActionListener {

   Button b;

   public void init()

      // Init() Method Invoke

   {

      b=new Button();

      b.setLabel("Start");

      b.addActionListener(this);

      add(b);

   }

   @Override

   public void actionPerformed(ActionEvent e) {


      if(b.getLabel()== "Start")

      {

         b.setLabel("Stop");

      }

      else
```

```
        {
            b.setLabel("Start");
        }
    }
}
```

Output:



Practical 6

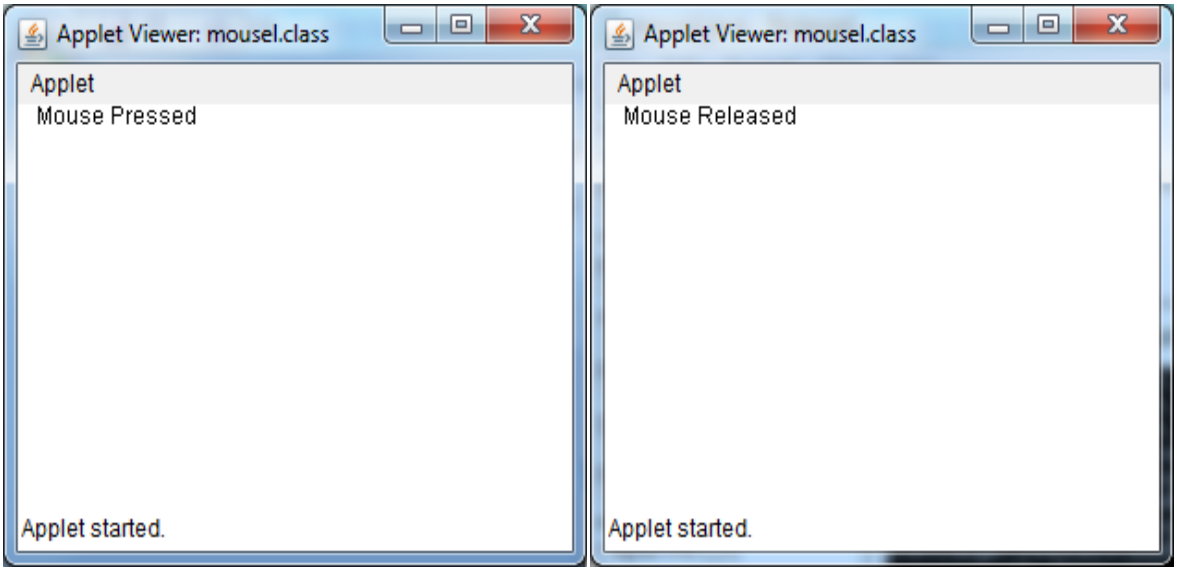
Aim :- Develop an applet that uses the mouse listener, which overrides only two methods, which are mousePressed and mouseReleased.

Program :

```
package com.company;
import java.applet.Applet;
import java.awt.*;
import java.awt.event.MouseEvent;
import java.awt.event.MouseListener;
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
public class Practical6 extends Applet implements MouseListener{

   String msg="";
   public void init()
         // Init() Method Invoked
   {
      setSize(400,400);
      addMouseListener(this);
   }
   public void paint(Graphics s)
   {
      showStatus(msg);
   }

   @Override
   public void mouseClicked(MouseEvent e) {

   }

   @Override
   public void mousePressed(MouseEvent e) {
      msg="Mouse Pressed";
      repaint();
   }

   @Override
   public void mouseReleased(MouseEvent e) {
      msg="Mouse Released";
```

```
        repaint();
    }

    @Override
    public void mouseEntered(MouseEvent e) {

    }

    @Override
    public void mouseExited(MouseEvent e) {

    }
}
```

Output:



Practical 7

Aim :- Develop a program that has only one button in the frame, clicking on the button cycles through the colors Red > Green > Blue and so on. One color changes per click. (use getBackGround() method to get the current color.)

Program :

```
package com.company;

import java.awt.*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.WindowAdapter;

import java.awt.event.WindowEvent;


public class Practical7 extends Frame implements ActionListener {

   Button btnColor=new Button("Click to change");

   Practical7()

   {

      setLayout(new FlowLayout());

      add(btnColor);

      btnColor.addActionListener(this);

      setVisible(true);

      setSize(800,500);

      addWindowListener(new WindowAdapter() {

         @Override

         public void windowClosing(WindowEvent e) {

            dispose();

         }
```

```
    });
  }

  @Override
  public void actionPerformed(ActionEvent e) {
    Color c=getBackground();
    if(c.equals(Color.white))
    {
      setBackground(Color.red);
    }
    else if(c.equals(Color.red))
    {
      setBackground(Color.green);
    }
    else if(c.equals(Color.green))
    {
      setBackground(Color.blue);
    }
    else
    {
      setBackground(Color.red);
    }
  }

  public static void main(String[] args) {
    new Practical7();
```

```
    }
}
```

Output:

Practical 8

Aim :- Develop an program that contains three check boxes and 30 x 30 pixel canvas.The three checkboxes should be labeled “Red”, “Green”,”Blue”. The selection of the check boxes determine the color of the canvas. For example, if the user selects both “Red” and “Blue”, the canvas should be purple.

Program :

```
package com.company;
import java.awt.*;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.awt.event.WindowAdapter;
import java.awt.event.WindowEvent;

public class Practical8 implements ItemListener {
   Frame f;
   Checkbox c1;
   Checkbox c2;
   Checkbox c3;
   Canvas c;
   int red = 0,blue=0,green=0;
   public Practical8()
   {
      f = new Frame("Practical - 8") ;
      c1 = new Checkbox("Red");
      c2 = new Checkbox("Green");
      c3 = new Checkbox("Blue");
      c1.addItemListener(this);
      c2.addItemListener(this);
      c3.addItemListener(this);

      c = new Canvas();
      c.setSize(30,30);
      c.setBackground(Color.BLACK);
      f.setLayout(new FlowLayout());
      f.add(c1);
      f.add(c2);
      f.add(c3);
```

```
        f.add(c);
        f.setSize(300,300);
        f.setVisible(true);
        f.setLayout(null);
        f.addWindowListener(new WindowAdapter() {
            @Override
            public void windowClosing(WindowEvent e) {
                f.dispose();
            }
        });
    }

    public static void main(String[] args)
    {
        Practical8 f = new Practical8();
    }
    public void itemStateChanged(ItemEvent ie)
    {
        if(c1.getState())
            red = 255;
        else
            red = 0;
        if(c2.getState())
            green = 255;
        else
            green = 0;
        if(c3.getState())
            blue = 255;
        else
            blue = 0;
        Color clr = new Color(red,green,blue);
        c.setBackground(clr);
    }
}
```

Output :

Practical 9

Aim : Create an application that displays a frame with a menu bar. When a user selects any menu or menu item, display that selection on a text area in the center of the frame.

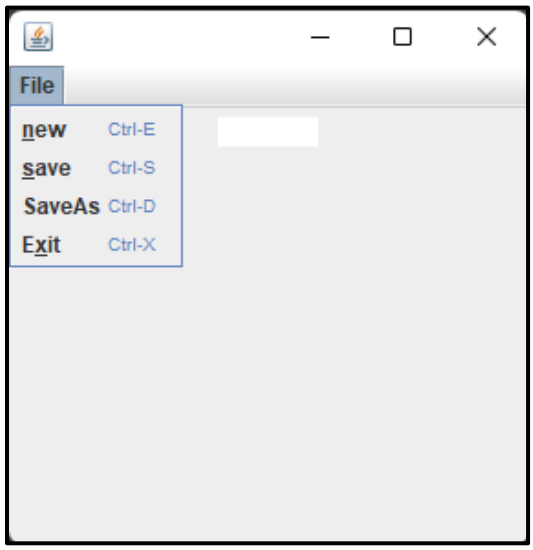
Program :

```
package com.company;
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class Practical9 extends JFrame implements ActionListener {
   private JMenuItem newOne;
   private JMenuItem save;
   private JMenuItem SaveAs;
   private JMenuItem exit;
   private JTextArea l;
   public Practical9()
   {
      FlowLayout f=new FlowLayout();

      JMenuBar menubar=new JMenuBar();
      JMenu menu=new JMenu("File");
      l=new JTextArea(null,1,5);

      JPanel p=new JPanel();
      add(p);

      l.setLayout(new FlowLayout());
      p.add(l);



      newOne=new JMenuItem("new");
      save=new JMenuItem("save");
      SaveAs=new JMenuItem("SaveAs");
      exit=new JMenuItem("Exit");


      newOne.setMnemonic('n');
      save.setMnemonic('s');
      SaveAs.setMnemonic('d');
```

```
    exit.setMnemonic('x');

    newOne.setAccelerator(KeyStroke.getKeyStroke('E',Event.CTRL_MASK));
    save.setAccelerator(KeyStroke.getKeyStroke('S',Event.CTRL_MASK));
    SaveAs.setAccelerator(KeyStroke.getKeyStroke('D',Event.CTRL_MASK));
    exit.setAccelerator(KeyStroke.getKeyStroke('X',Event.CTRL_MASK));


    menubar.add(menu);
    menu.add(newOne);
    menu.add(save);
    menu.add(SaveAs);
    menu.add(exit);

    newOne.addActionListener(this);
    save.addActionListener(this);
    SaveAs.addActionListener(this);
    exit.addActionListener(this);




    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setJMenuBar(menubar);
    setSize(300,300);
    //setLayout(null);
    setVisible(true);
  }

  public void actionPerformed(ActionEvent e)
  {
    if(e.getSource()== newOne)
    {
      l.setText("new");
    }
    else if(e.getSource()== save)
    {
      l.setText("save");
    }
    else if(e.getSource()== SaveAs)
    {
      l.setText("saveAS");
    }
    else if(e.getSource()== exit)
```

```
      {
        l.setText("exit");
      }

    }

    public static void main(String[] args) {
      new Practical9();
    }
}
```

Output:



Practical 10

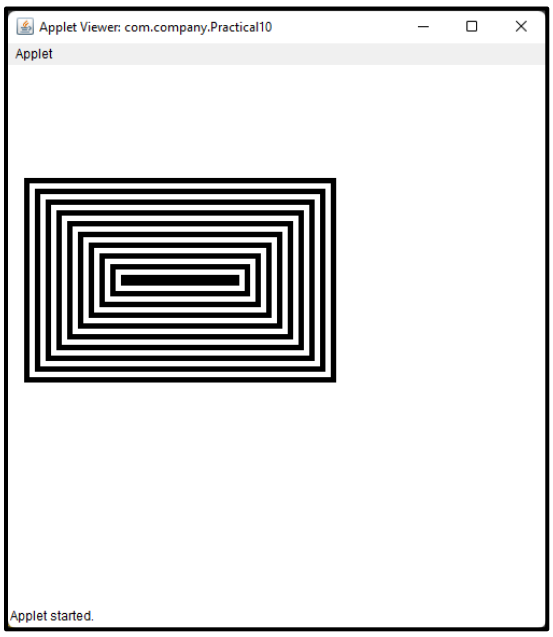
Aim :- Develop a program that draws two sets of ever-decreasing rectangles one in outline form and one filled alternately in black and white.

Program :

```
package com.company;

import java.applet.Applet;

import java.awt.*;

import java.awt.Graphics;

import java.awt.Color;

public class Practical10 extends Applet {

   int x, y, h, w;

   @Override

   public void init() {

      // TODO Auto-generated method stub

      super.init();

      setSize(500, 500);

      x = 10;

      y = 100;

      h = 300;

      w = 200;

   }

   @Override

   public void paint(Graphics g) {

      // TODO Auto-generated method stub

      super.paint(g);

      for (int i = 0; i < 20; i++) {
```

```
            x = x + 5;
            y = y + 5;
            h = h - 10;
            w = w - 10;
            if (i % 2 == 0) {
               g.setColor(Color.black);
               g.fillRect(x, y, h, w);
            } else {
               g.setColor(Color.white);
               g.fillRect(x, y, h, w);
            }
         }
      }
   }
}
```

Output:



Practical 11

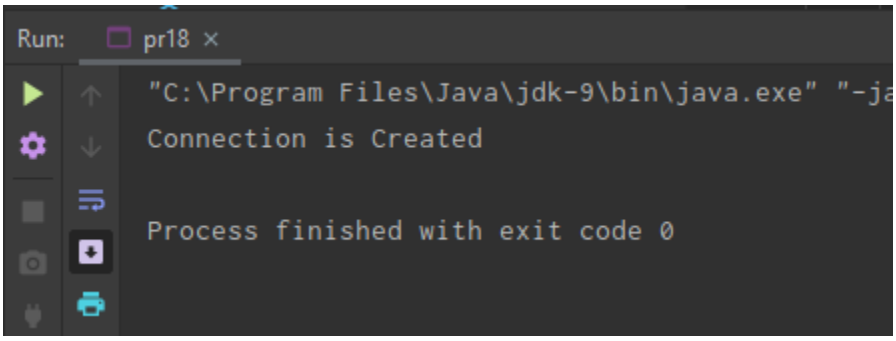
Aim : Develop a database application that uses any JDBC driver

Program :

```
package com.company;

import java.sql.*;

public class pr11 {

   // Practical 11

   public static void main(String args[]){

      try{

         Class.forName("com.mysql.cj.jdbc.Driver");


         String url = "jdbc:mysql://localhost:3306/smobile";

         String username = "root";

         String password = "";

         Connection con = DriverManager.getConnection(url,username,password);

         if(con.isClosed()){

            System.out.println("Connection is Closed");

         }

         else{

            System.out.println("Connection is Created");

         }

      }

      catch (SQLException throwables) {

         throwables.printStackTrace();
```

```
        }
        catch (ClassNotFoundException e) {
            e.printStackTrace();
        }
    }
}
```

Output :



Practical 12

Aim : Develop a Graphical User Interface that performs the following SQL Operations: a). Insert b).Delete c). Update

Program :

```
import java.util.logging.Level;
import java.util.logging.Logger;
import java.sql.*;
import javax.swing.JOptionPane;
public class Product extends javax.swing.JFrame {
   public Product() {
      initComponents();
      Connect();
      LoadProductNo();
   }
   Connection con;
   PreparedStatement pst;
   ResultSet rs;
   public void Connect(){
      try {
         Class.forName("com.mysql.cj.jdbc.Driver");
         con = DriverManager.getConnection("jdbc:mysql://localhost:3306/smobile",
"root","");
      } catch (ClassNotFoundException ex) {
         Logger.getLogger(Product.class.getName()).log(Level.SEVERE, null, ex);
      } catch (SQLException ex) {
         Logger.getLogger(Product.class.getName()).log(Level.SEVERE, null, ex);
```

```
    }
}
public void LoadProductNo(){
    try {
        pst = con.prepareStatement("select id from products");
        rs = pst.executeQuery();
        txtpid.removeAllItems();
        while(rs.next()){
            txtpid.addItem(rs.getString(1));
        }
    }
    catch (SQLException ex) {
        Logger.getLogger(Product.class.getName()).log(Level.SEVERE, null, ex);
    }
}
@SuppressWarnings("unchecked")
private void initComponents() {
    jLabel1 = new javax.swing.JLabel();
    jPanel1 = new javax.swing.JPanel();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jLabel4 = new javax.swing.JLabel();
    txtpname = new javax.swing.JTextField();
    txtprice = new javax.swing.JTextField();
    txtqty = new javax.swing.JTextField();
    jLabel5 = new javax.swing.JLabel();
```

```
        txtpid = new javax.swing.JComboBox<>();
        jButton4 = new javax.swing.JButton();
        jPanel2 = new javax.swing.JPanel();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton5 = new javax.swing.JButton();
        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        jLabel1.setFont(new java.awt.Font("Segoe UI", 1, 36)); // NOI18N
        jLabel1.setText("Product Information");
        jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder("Product
Information"));
        jLabel2.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N
        jLabel2.setText("Product name :");
        jLabel3.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N
        jLabel3.setText("Price : ");
        jLabel4.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N
        jLabel4.setText("Qty :");
        jLabel5.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N
        jLabel5.setText("Product ID :");
        jButton4.setText("Search");
        jButton4.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton4ActionPerformed(evt);
            }
        });
        javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
```

```
        jPanel1.setLayout(jPanel1Layout);
        jPanel1Layout.setHorizontalGroup(
            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(23, 23, 23)
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(jLabel2)
                    .addComponent(jLabel3)
                    .addComponent(jLabel4))
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING
, false)
                    .addComponent(txtprice, javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(txtpname, javax.swing.GroupLayout.Alignment.LEADING)
                    .addComponent(txtqty, javax.swing.GroupLayout.DEFAULT_SIZE, 147,
Short.MAX_VALUE))
                .addGap(27, 27, 27)
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
                    .addComponent(jLabel5, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addComponent(txtpid, 0, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                    .addComponent(jButton4, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        );
        jPanel1Layout.setVerticalGroup(
```

```
        jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addContainerGap()
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE
)
                .addComponent(jLabel2)
                .addComponent(txtpname, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jLabel5))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
                .addComponent(txtpid)
                .addComponent(txtprice)
                .addComponent(jLabel3, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE
)
                .addComponent(txtqty, javax.swing.GroupLayout.PREFERRED_SIZE, 25,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addComponent(jButton4))
            .addComponent(jLabel4))
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
    );
```

```
        jPanel2.setBorder(new
javax.swing.border.SoftBevelBorder(javax.swing.border.BevelBorder.RAISED));
        jButton1.setText("Add");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });
        jButton2.setText("Update");
        jButton2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton2ActionPerformed(evt);
            }
        });
        jButton3.setText("Delete");
        jButton3.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton3ActionPerformed(evt);
            }
        });
        jButton5.setText("New");
        jButton5.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton5ActionPerformed(evt);
            }
        });
        javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);
```

```
        jPanel2.setLayout(jPanel2Layout);
        jPanel2Layout.setHorizontalGroup(
            jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addContainerGap()
                .addComponent(jButton1, javax.swing.GroupLayout.PREFERRED_SIZE, 82,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(42, 42, 42)
                .addComponent(jButton2, javax.swing.GroupLayout.PREFERRED_SIZE, 85,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 52,
Short.MAX_VALUE)
                .addComponent(jButton3, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(45, 45, 45)
                .addComponent(jButton5, javax.swing.GroupLayout.PREFERRED_SIZE, 75,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(25, 25, 25))
        );
        jPanel2Layout.setVerticalGroup(
            jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel2Layout.createSequentialGroup()
                .addContainerGap()

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE
)
                    .addComponent(jButton1)
                    .addComponent(jButton2)
                    .addComponent(jButton3)
```

```
            .addComponent(jButton5))
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
    );
    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
      layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
      .addGroup(layout.createSequentialGroup()
        .addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addGroup(layout.createSequentialGroup()
              .addComponent(jLabel1)
              .addGap(0, 0, Short.MAX_VALUE))
            .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addContainerGap())
    );
    layout.setVerticalGroup(
      layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
      .addGroup(layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jLabel1)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
```

```
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jPanel2, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        );
        pack();
        setLocationRelativeTo(null);
    }
    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
            String pname = txtpname.getText();
            String price = txtprice.getText();
            String qty = txtqty.getText();
            String pid = txtpid.getSelectedItem().toString();
            pst = con.prepareStatement("update products name set pname=?, price=?, qty=?
where id = ?");
            pst.setString(1, pname);
            pst.setString(2, price);
            pst.setString(3, qty);
            pst.setString(4, pid);
            int k = pst.executeUpdate();
            if(k==1){
               JOptionPane.showMessageDialog(this, "Record Updated");
               txtpname.setText("");
               txtprice.setText("");
               txtqty.setText("");
               txtpname.requestFocus();
```

```
            LoadProductNo();
        }
        else{
            JOptionPane.showMessageDialog(this, "Record Can't be Updated");
        }
            }
    catch (SQLException ex) {
        Logger.getLogger(Product.class.getName()).log(Level.SEVERE, null, ex);
    }
}
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        String pname = txtpname.getText();
        String price = txtprice.getText();
        String qty = txtqty.getText();
        pst = con.prepareStatement("insert into products(pname, price, qty)values(?,?,?)");
        pst.setString(1, pname);
        pst.setString(2, price);
        pst.setString(3, qty);
        int k = pst.executeUpdate();
        if(k==1){
            JOptionPane.showMessageDialog(this, "Record Added");
            txtpname.setText("");
            txtprice.setText("");
            txtqty.setText("");
            txtpname.requestFocus();
```

```
            LoadProductNo();
        }
        else{
            JOptionPane.showMessageDialog(this, "Record Can't be Add");
        }
              }
    catch (SQLException ex) {
        Logger.getLogger(Product.class.getName()).log(Level.SEVERE, null, ex);
    }
}
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
    String pid = txtpid.getSelectedItem().toString();

    try {
        pst = con.prepareStatement("select * from products where id = ?");
        pst.setString(1, pid);
        rs = pst.executeQuery();
        if(rs.next()==true){
            txtpname.setText(rs.getString(2));
            txtprice.setText(rs.getString(3));
            txtqty.setText(rs.getString(4));
        }
    } catch (SQLException ex) {
        Logger.getLogger(Product.class.getName()).log(Level.SEVERE, null, ex);
    }}
private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
```

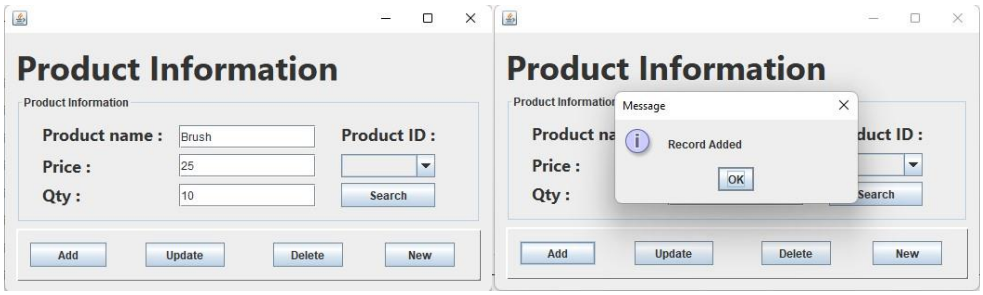
```
    txtpname.setText("");
    txtprice.setText("");
    txtqty.setText("");
    txtpname.requestFocus();
    LoadProductNo();
}
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
        String pid = txtpid.getSelectedItem().toString();
        pst = con.prepareStatement("delete from products where id = ?");
        pst.setString(1, pid);

        int k = pst.executeUpdate();
        if(k==1){
            JOptionPane.showMessageDialog(this, "Record Deleted");
            txtpname.setText("");
            txtprice.setText("");
            txtqty.setText("");
            txtpname.requestFocus();
        }
        else{
            JOptionPane.showMessageDialog(this, "Record Can't be deleted");
        }
            }
    catch (SQLException ex) {
        Logger.getLogger(Product.class.getName()).log(Level.SEVERE, null, ex);
```

```
    }}
  public static void main(String args[]) {
    try {
      for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
        if ("Metal".equals(info.getName())) {
          javax.swing.UIManager.setLookAndFeel(info.getClassName());
          break;
        }
      }
    } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Product.class.getName()).log(java.util.logging.Level.SEVER
E, null, ex);
    } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Product.class.getName()).log(java.util.logging.Level.SEVER
E, null, ex);
    } catch (IllegalAccessException ex) {
java.util.logging.Logger.getLogger(Product.class.getName()).log(java.util.logging.Level.SEVER
E, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
java.util.logging.Logger.getLogger(Product.class.getName()).log(java.util.logging.Level.SEVER
E, null, ex);
    }
    java.awt.EventQueue.invokeLater(new Runnable() {
      public void run() {
        new Product().setVisible(true);
      }
```

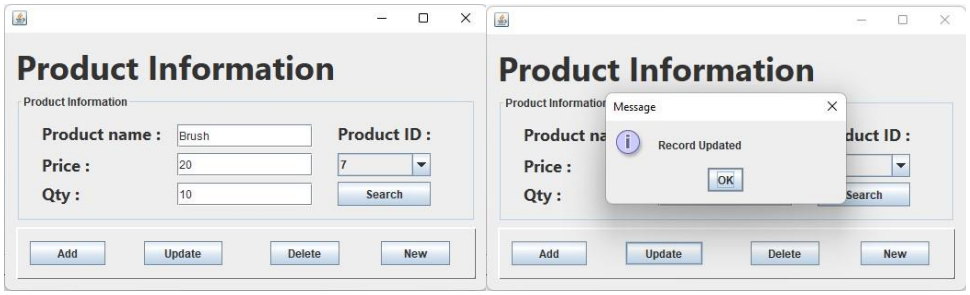
```
    });
  }
  private javax.swing.JButton jButton1;
  private javax.swing.Jbutton jButton2;
  private javax.swing.Jbutton jButton3;
  private javax.swing.Jbutton jButton4;
  private javax.swing.Jbutton jButton5;
  private javax.swing.Jlabel jLabel1;
  private javax.swing.Jlabel jLabel2;
  private javax.swing.Jlabel jLabel3;
  private javax.swing.Jlabel jLabel4;
  private javax.swing.Jlabel jLabel5;
  private javax.swing.Jpanel jPanel1;
  private javax.swing.Jpanel jPanel2;
  private javax.swing.JcomboBox<String> txtpid;
  private javax.swing.JtextField txtpname;
  private javax.swing.JtextField txtprice;
  private javax.swing.JtextField txtqty;
}
```

Output:

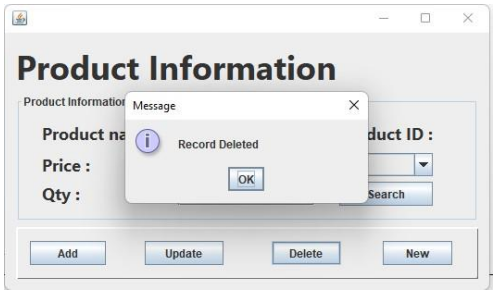
Record Add



For Update



For Delete



Practical 13

Aim : Develop a program to present a set of choice for user to select a product and display the price of product.

Program :

GUIList.java

```
package com.company;

import java.awt.BorderLayout;
import java.awt.Color;
import java.awt.FlowLayout;
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;

/**
 *
 * Practical 13 First File
 */
public class GUIList  extends JFrame implements ItemListener{

   JComboBox combo;

   JPanel p,p2;
   JLabel Id,IdValue,Dep,DepValue;

   public String names[]={" ","Laptop","Desktop","Keyboard","Mouse"};
   DataDB db=new DataDB();

   GUIList()
```

```
{
   p=new JPanel();
   combo=new JComboBox(names);
   p.add(combo);

   p2=new JPanel();
   Id =new JLabel("ID:");
   IdValue=new JLabel("");
   Dep=new JLabel("Department:");
   DepValue=new JLabel("");

   p2.add(Id);
   p2.add(IdValue);
   p2.add(Dep);
   p2.add(DepValue);

   combo.addItemListener(this);
   add(p);
   add(p2);
   setLayout(new FlowLayout());
   setDefaultCloseOperation(EXIT_ON_CLOSE);
   setSize(500,500);
   setVisible(true);
}
public static void main(String args[])
{
   GUIList gui=new GUIList();
}
@Override
public void itemStateChanged(ItemEvent e) {
   String Name=(String) combo.getItemAt(combo.getSelectedIndex());

   try {
      db.getData(Name);
      IdValue.setText(String.valueOf(DataDB.id));
      DepValue.setText(String.valueOf(DataDB.p_price));
   } catch (ClassNotFoundException ex) {
      Logger.getLogger(GUIList.class.getName()).log(Level.SEVERE, null, ex);
   } catch (SQLException ex) {
      Logger.getLogger(GUIList.class.getName()).log(Level.SEVERE, null, ex);
   }

}
```

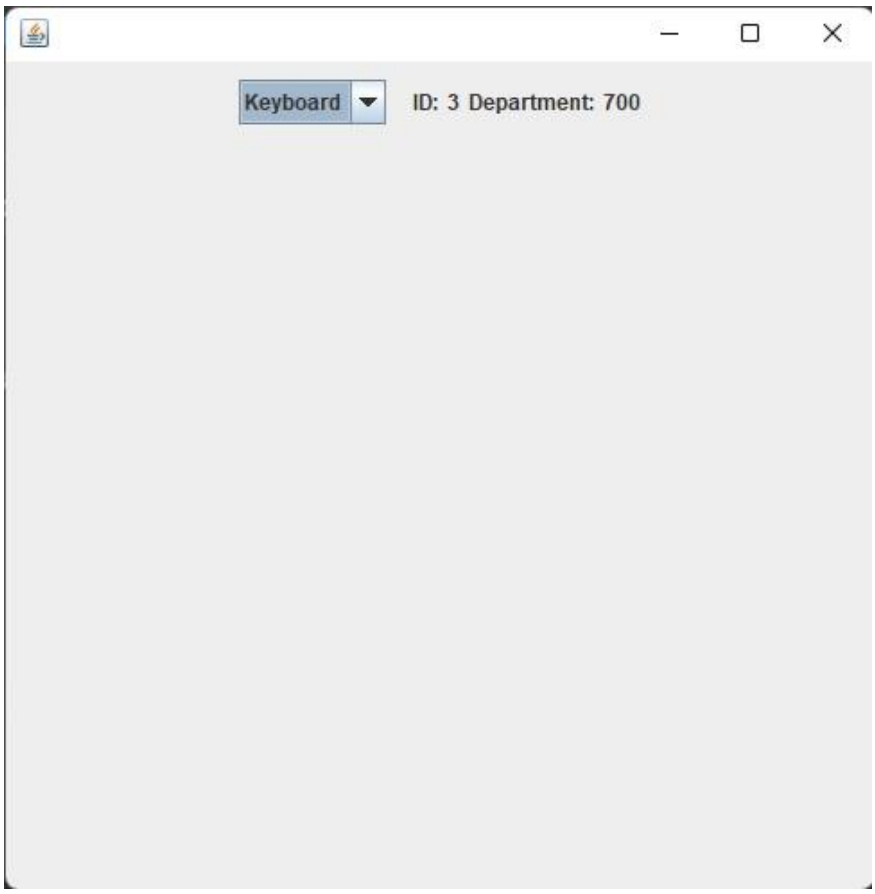
```
}
```

DataDB.java

```
package com.company;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

/**
 *
 * Practical 13 Second File
 */
public class DataDB {

   public static int id=0;
   public static int p_price=0;

   DriverClass d=new DriverClass();

   public void getData(String Name) throws ClassNotFoundException, SQLException
   {
      Class.forName(d.Driver);
      Connection con=DriverManager.getConnection(d.Path,d.UserName,d.password);
      System.out.println("Connection Establish");
      PreparedStatement ps=con.prepareStatement("Select * from product where p_name=?
");
      ps.setString(1, Name);
      ResultSet rs=ps.executeQuery();

      while(rs.next())
      {
         id=rs.getInt("id");
         p_price=rs.getInt("p_price");
      }
   }

}
```

DriverClass.java

```
package com.company;

public class DriverClass {

   public String Driver="com.mysql.cj.jdbc.Driver";
   public String UserName="root";
   public String password="";
   public String Path="jdbc:mysql://localhost:3306/student";
}
```



Practical 14

Aim : Develop a simple servlet program which maintains a counter for the number of times it has been accessed since its loading, initialize the counter using deployment descriptor.

Program :

HitCounter.java

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.*;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class HitCounter extends HttpServlet
{
        int c;
        public void init()
        {
                ServletConfig s=getServletConfig();
                c=Integer.parseInt(s.getInitParameter("counter"));
        }
        public void doGet(HttpServletRequest req, HttpServletResponse res) throws
        ServletException, IOException
        {
                c++;
                PrintWriter out = res.getWriter();
                out.println("Total Hit: " + c);
        }
}
```

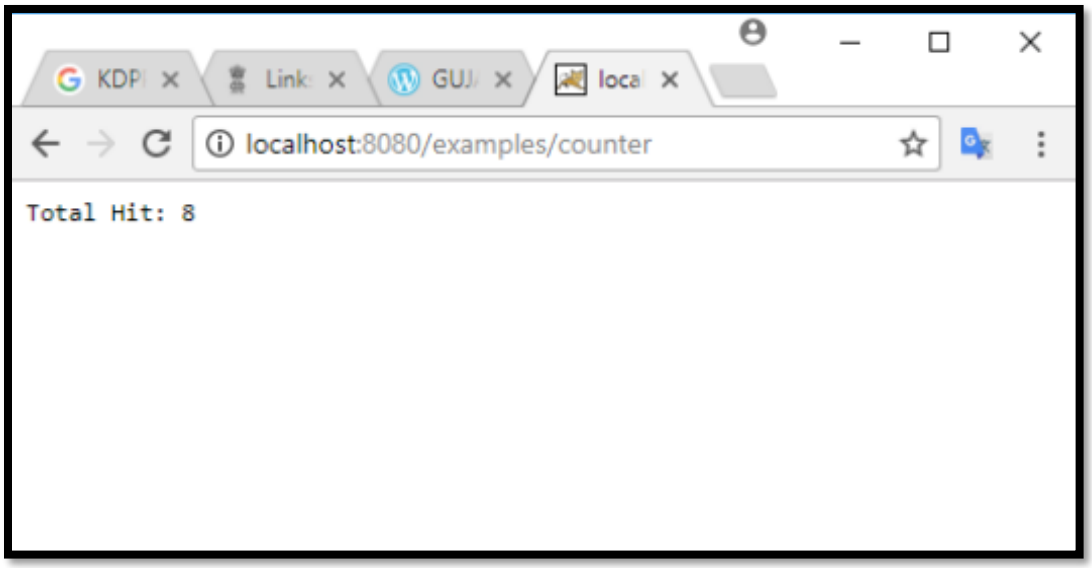
Web.xml

```
<servlet>
<servlet-name>pagecounter</servlet-name>
 <servlet-class>HitCounter</servlet-class>

 <init-param>
<param-name>counter</param-name>
<param-value>0</param-value>
```

</init-param>
</servlet>
<servlet-mapping>
<servlet-name>pagecounter</servlet-name>
<url-pattern>/counter</url-pattern>
</servlet-mapping>

Compile:
javac –classpath C:\users\mayur\desktop\apache-tomcat-9.0.1\lib\servlet-api.jar HitCounter.java

Output :



Practical 15

Aim :- Create a web form which processes servlet and demonstrates use of sessions.

Program :

login.html :

```
<html>
<head>
<title> Login Page </title>
</head>
<body>
<form action="loginservlet" method="post" >
        <table>
        <tr>
                <td>User Name: </td>
                <td> <input type = "text" name = "name"> </td>
        </tr>
        <tr>
                <td>Password: </td>
                <td>
                        <input type = "password" name = "pass">
                </td>
        </tr>
        <tr>
                <td></td>
                <td>
                        <input type = "submit" name = "submit" value = "SUBMIT">
                </td>
        </tr>
</form>
</body>
</html>
```

loginservlet.java

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
```

```
import javax.servlet.http.HttpSession;
public class loginservlet extends HttpServlet
{
   protected void doPost (HttpServletRequest req, HttpServletResponse   res) throws
   ServletException, IOException {
   res.setContentType("text/html");
   PrintWriter out=res.getWriter();
   String name=req.getParameter("name");
   String password=req.getParameter("pass");
if(name.equals("Mayur") && password.equals("admin"))
{
        HttpSession session=req.getSession();
        session.setAttribute("Name", name);
        out.println("<br/><h1>Welcome :"+ name + "</h1> "); out.println("<br/> <a
href=profile> Click here </a>");
}
else
{
        out.println("<h1>You Have entered Wrong Password </h1>"); out.println("<br/> <a
        href=login.html> Click here </a> to Login");  }
}
}
```

Profile.java

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
public class profile extends HttpServlet
{
  protected void doGet (HttpServletRequest req, HttpServletResponse   res) throws
ServletException, IOException  {

     res.setContentType("text/html");
     PrintWriter out=res.getWriter();
```

```
      HttpSession session=req.getSession(false);
      if(session!=null)
      {
         String name=(String) session.getAttribute("Name");  out.println("Welcome," +
name);
         out.println("<h1>This is Session Program. </h1> ");
         out.println("<a href=logoutservlet> Click here to Logout </a>");
      }
      else
      {
         out.println("<br> <h1> Plz Login First</h1>");
         out.println("<br/> <a href=login.html> Click here </a> to Login");
      }
   }
}
```

Logoutservlet.java

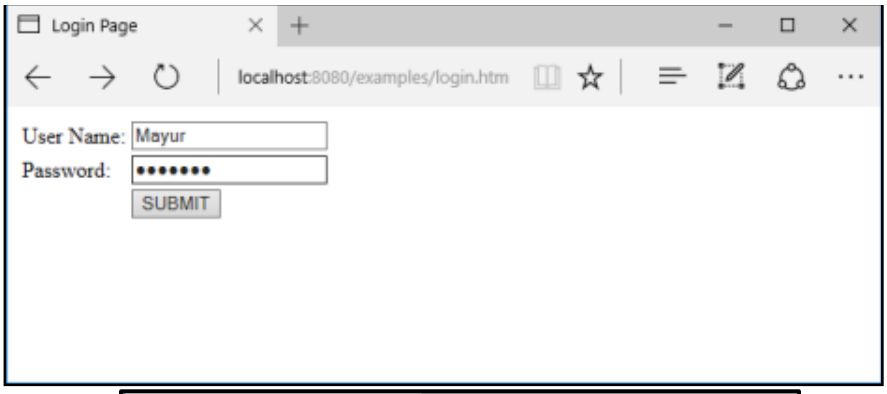
```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;
public class logoutServlet extends HttpServlet
{
 public void doGet (HttpServletRequest req,HttpServletResponse res)   throws IOException,
ServletException  {
 res.setContentType("text/html");
 PrintWriter out=res.getWriter();

 HttpSession session=req.getSession();
 session.invalidate();

 out.println("<br> <h1>You have successfully Logout </h1>");  out.println("<br/> <a
href=login.html> Click here </a> to Login");  } }
```

Web.xml

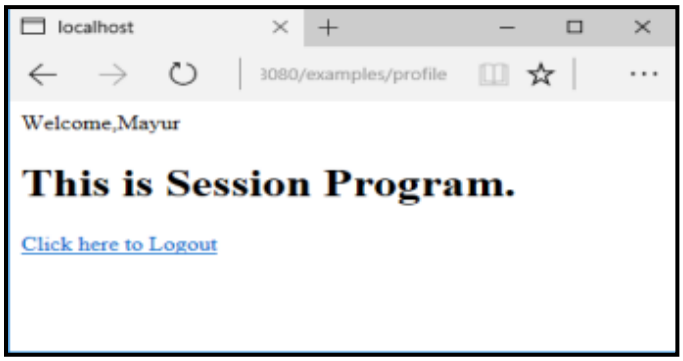
```
<servlet>
 <servlet-name>loginservlet</servlet-name>  <servlet-
class>loginservlet</servlet-class>  </servlet>


<servlet>
 <servlet-name>logoutservlet</servlet-name>  <servlet-
class>logoutServlet</servlet-class>  </servlet>


<servlet>
 <servlet-name>profile</servlet-
name>  <servlet-class>profile</servlet-
class>  </servlet>


<servlet-mapping>
 <servlet-name>profile</servlet-name>  <url-
pattern>/profile</url-pattern>  </servlet-
mapping>

<servlet-mapping>
 <servlet-name>logoutservlet</servlet-name>  <url-
pattern>/logoutservlet</url-pattern>  </servlet-
mapping>

<servlet-mapping>
 <servlet-name>loginservlet</servlet-name>  <url-
pattern>/loginservlet</url-pattern>  </servlet-
mapping>
```

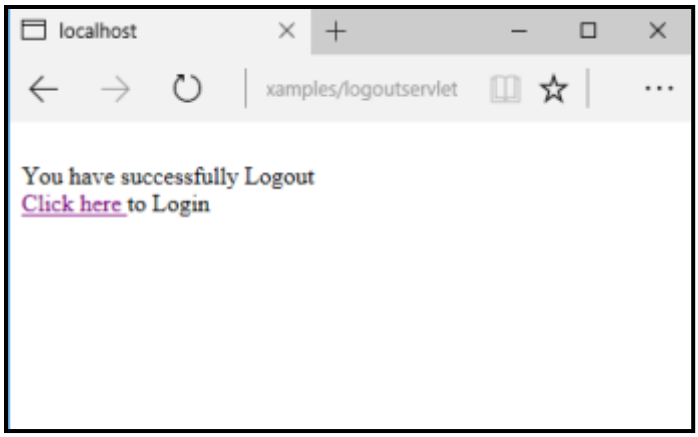
Output :

Note : here username : Mayur & password : admin


Login Page
localhost:8080/examples/login.htm
User Name: Mayur
Password:
SUBMIT


localhost
alhost:8080/examples/
Welcome :Mayur
Click here


localhost
3080/examples/profile
Welcome,Mayur
This is Session Program.
Click here to Logout


localhost
xamples/logoutservlet
You have successfully Logout
Click here to Login

Program 16

Aim : Develop a simple JSP program for user registration and then control will be transfer it into second page.

Program :

Firstpage.jsp.

```
<html>
    <head>
        <title>Registration Page</title>
    </head>
<body>
    <form action="secondpage.jsp" method="post">
            Enter your UserName :
                <input type="text" name="name"> <br>
            Enter your Password :
                <input type="password" name="pas"> <br>
            Enter your email :
                <input type="email" name="email"> <br>
            <input type="submit">
    </form>
</body>
</html>
```

Secondpage.jsp

```
<html>
    <head>
     <title>Welcome Page</title>
    </head>

 <body>
    <%
        String username=request.getParameter("name");
        String password=request.getParameter("pas");
        String email=request.getParameter("email");
```

```
                out.print(username + ", registered successfully!!! ");
        %>
        </body>
</html>
```

Practical 17

Aim : Develop a simple JSP program for user login form with static and dynamic database

Program :

Login With Static Database :

Login.jsp

```
<html>
        <head>
                <title> Log in </title>
        </head>
        <body>
                 <form action="welcome.jsp" method="post">
                        Username :
                <input type="text" name="uname"> <br/>
                Password :
                <input type="password" name="pass"> <br/>
                <input type="submit">
                 </form>
         </body>
</html>
```

Welcome.jsp

```
<%
        String username=request.getParameter("uname");
        String password=request.getParameter("pass");
        if(username.equals("jenish") && password.equals("jadav")) {
                out.print("Welcome, " + username + "!");
        }
        else
        {
                out.print("Invalid Username or Password. ")
        }
%>
```

Login With Dynamic Database :

Web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
        xmlns="http://java.sun.com/xml/ns/javaee"
        xmlns:web="http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
        xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
        http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd" id="WebApp_ID"
        version="2.5">
  <display-name>LoginFormStruts1</display-name>
  <servlet>
    <servlet-name>action</servlet-name>
    <servlet-class>org.apache.struts.action.ActionServlet</servlet-class>
    <init-param>
      <param-name>config</param-name>
      <param-value>
     /WEB-INF/struts-config.xml
    </param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
  </servlet>
 <servlet-mapping>
    <servlet-name>action</servlet-name>
    <url-pattern>*.do</url-pattern>
  </servlet-mapping>
```

```
    <welcome-file-list>
      <welcome-file>index.jsp</welcome-file>
    </welcome-file-list>
</web-app>
```

Loginform.java

```
package com.example.javawebtutor.form;
import javax.servlet.http.HttpServletRequest;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionMapping;
public class LoginForm extends ActionForm {
   private String userName = null;
   private String password = null;
   public String getUserName() {
      return userName;
   }
   public void setUserName(String userName) {
      this.userName = userName;
   }
   public String getPassword() {
      return password;
   }
   public void setPassword(String password) {
      this.password = password;
   }
```

```
    @Override
    public void reset(ActionMapping mapping, HttpServletRequest request) {
        this.password = null;
    }
}
```

LoginAction.java file:

```
package com.example.javawebtutor.action;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.apache.struts.action.Action;
import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionForward;
import org.apache.struts.action.ActionMapping;
import com.example.javawebtutor.form.LoginForm;
public class LoginAction extends Action {
    @Override
    public ActionForward execute(ActionMapping mapping, ActionForm form,
            HttpServletRequest request, HttpServletResponse response)
            throws Exception {
        LoginForm loginForm = (LoginForm) form;
        if (loginForm.getUserName() == null || loginForm.getPassword() == null
                || !loginForm.getUserName().equalsIgnoreCase("Mukesh")
                || !loginForm.getPassword().equals("kumar")) {
            return mapping.findForward("success");
```

```
    } else
        return mapping.findForward("failure");
  }
}
```

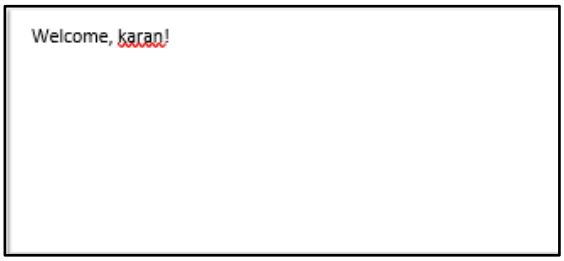
Login.jsp

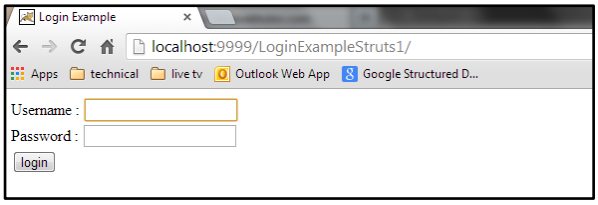
```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Login Example</title>
</head>
<body>
  <html:form action="/login" focus="userName">
  Username : <html:text property="userName" />
    <br>
  Password : <html:password property="password" />
    <br>
    <html:submit value="login" />
  </html:form>
</body>
</html>
```

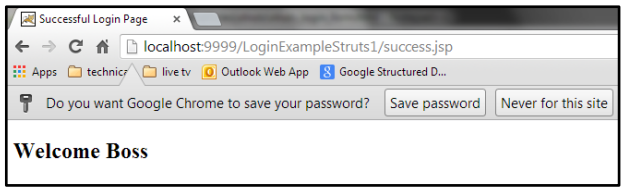
Success.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Successful Login Page</title>
</head>
<body>
    <h2>Welcome Boss</h2>
</body>
</html>
```

Output :







Practical 18

Aim : Develop a JSP program to display the grade of a student by accepting the marks of five subjects.

Program :

input.jsp

```
<html>
<head>
    <title>Subject Marks</title>
</head>
<body>
    <h1>Marks Entry of Semester - 6</h1>
    <form action="result.jsp" method="POST">
        Enter AJP Marks :
        <input type="text" name="AJP"><br><br>
        Enter NMA Marks :
        <input type="text" name="NMA"><br><br>
        Enter MCAD Marks :
        <input type="text" name="MCAD"><br><br>
        Enter PPUD Marks :
        <input type="text" name="PPUD"><br><br>
        Enter Project Marks :
        <input type="text" name="PRO"><br><br>
        <input type="submit">
    </form>
```

```
</body>

</html>
```

result.jsp

```
<%

    int AJP=Integer.parseInt(request.getParameter("AJP"));  int
    NMA=Integer.parseInt(request.getParameter("NMA"));  int
    MCAD=Integer.parseInt(request.getParameter("MCAD"));  int
    PPUD=Integer.parseInt(request.getParameter("PPUD"));  int
    PROJECT=Integer.parseInt(request.getParameter("PRO"));

    int Total = AJP+NMA+MCAD+PPUD+PROJECT;

    double avg = Total/5.0;

    if(avg > =90 )

    {

        out.println(" your grade is: A");

    }

    else if (avg >= 80)

    {

        out.println("your grade is: B");

    }

    else if (avg >= 70)

    {

        out.println("your grade is: C");

    }

    else if (avg >= 60)

    {

        out.println("your grade is: D");
```

```
        }
        else
        {
                out.println("your grade is: E");
        }
%>
```