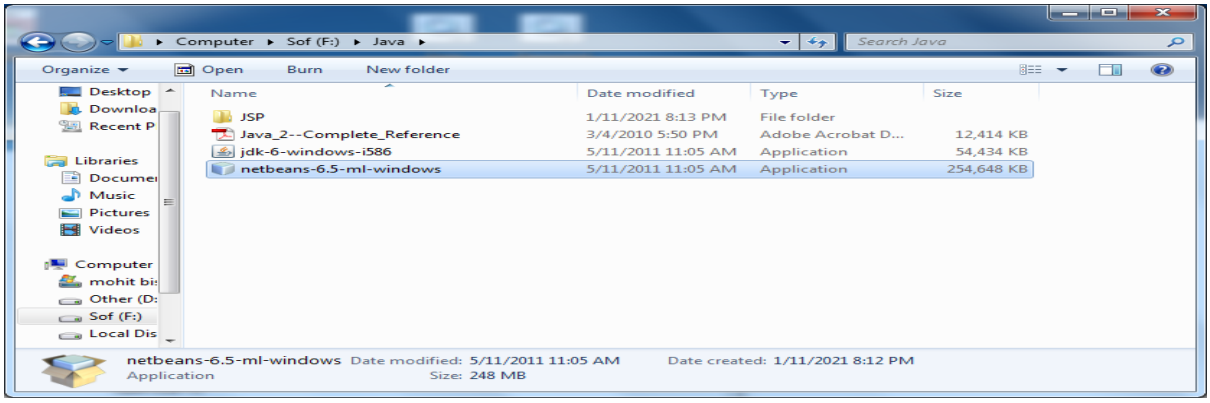
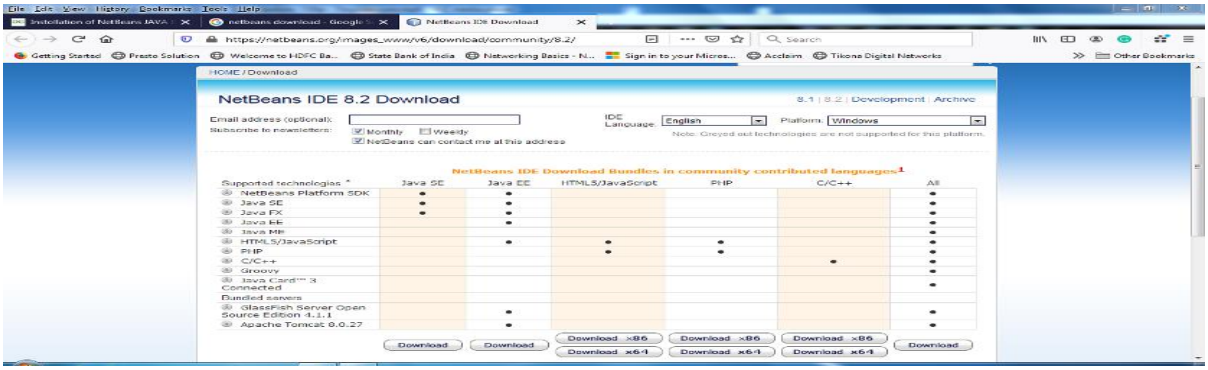
Practical – 1

AIM: Installation of NetBeans on Windows.

1. You need to have a setup file of the NetBeans JAVA into your setup.



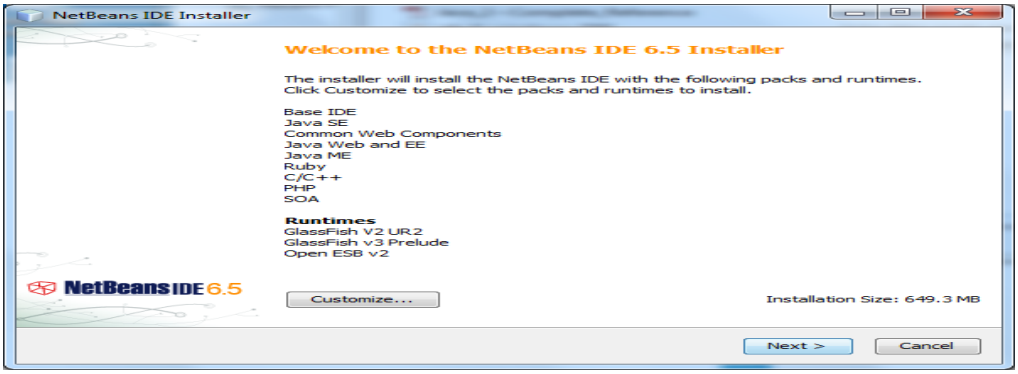
2. If you didn't have the setup you can download from the following link: https://netbeans.org/images_www/v6/download/community/8.2



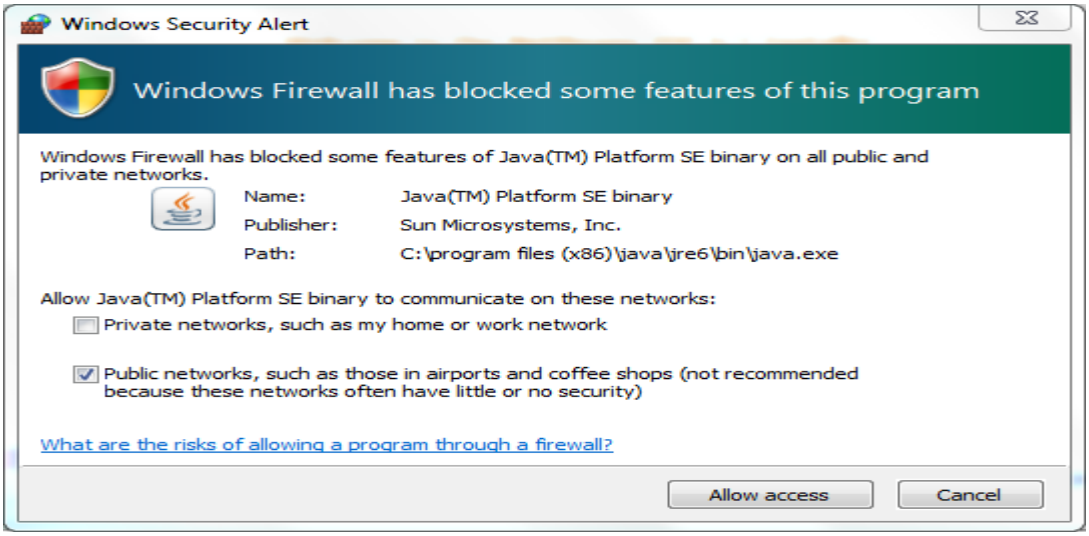
3. You can download any type of setup as per your requirements from the above mention web page.

4. Right-click on the setup or you can Double-Click on the setup by using the mouse.

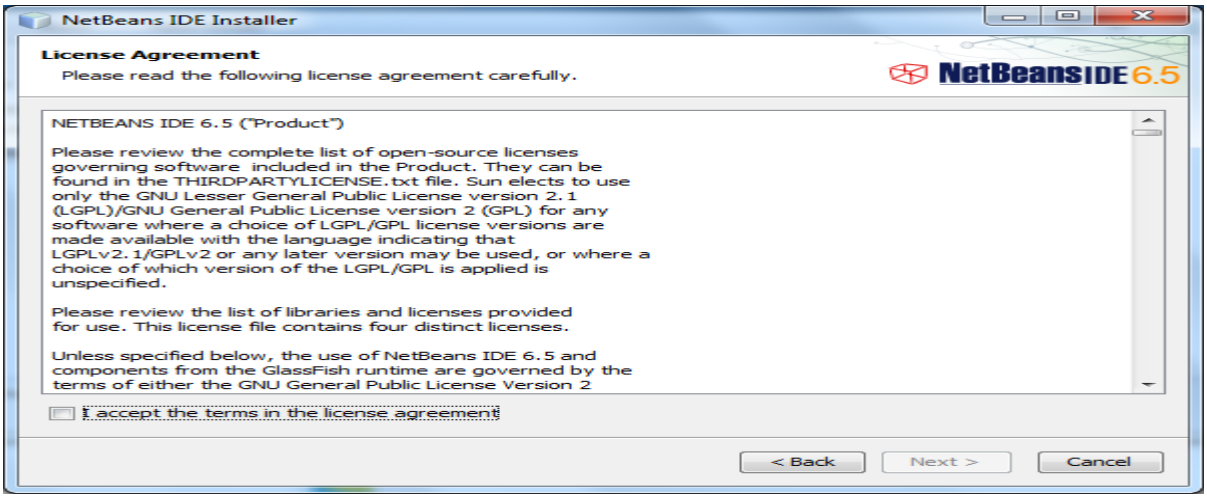
5. Click on the next option.



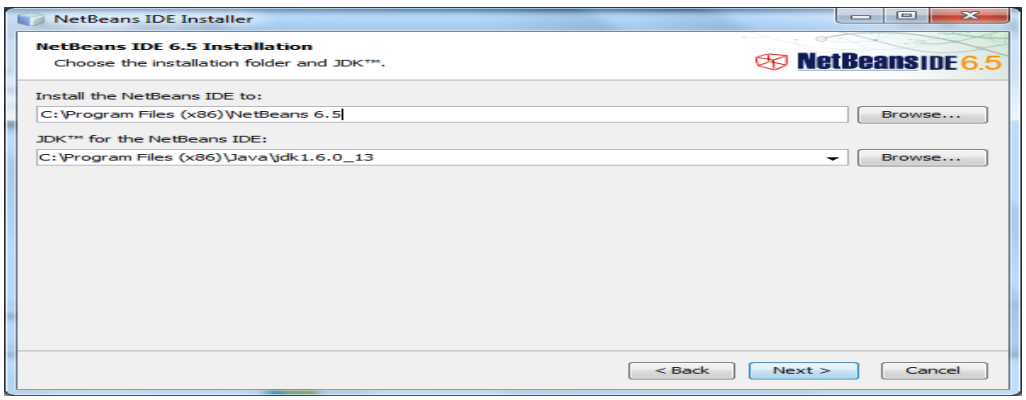
6. Check on the "**Private networks, such as my home and work network".**



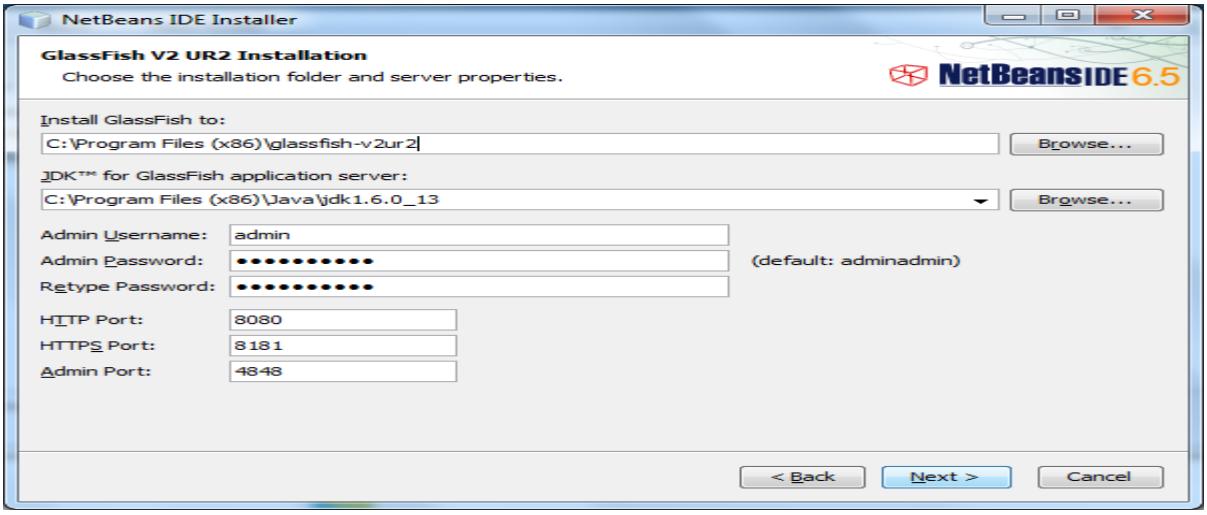
7. Click on the **Allow access** button.
8. Check on the **"I accept"** option and click on the "**Next"** button.



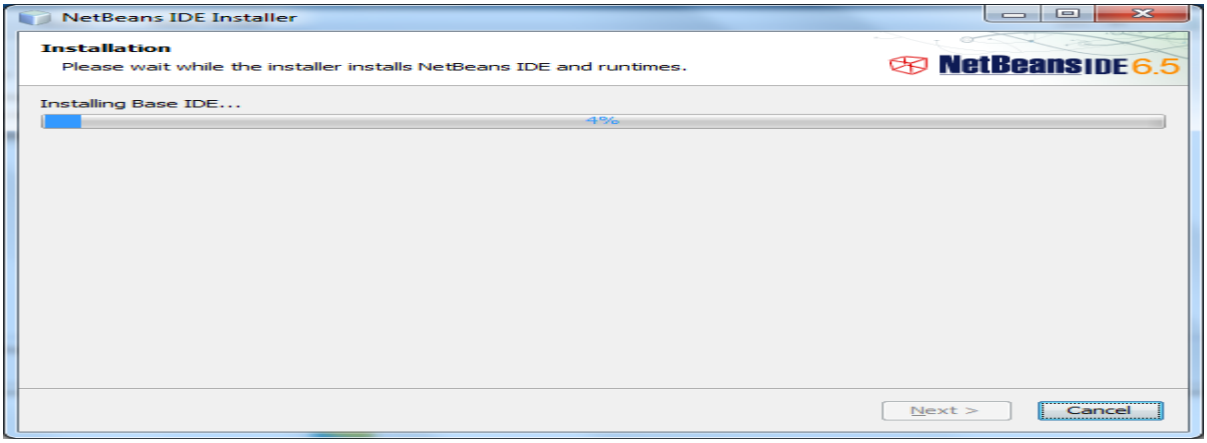
9. Select the path where you want to install the software and press the "**Next"** button.



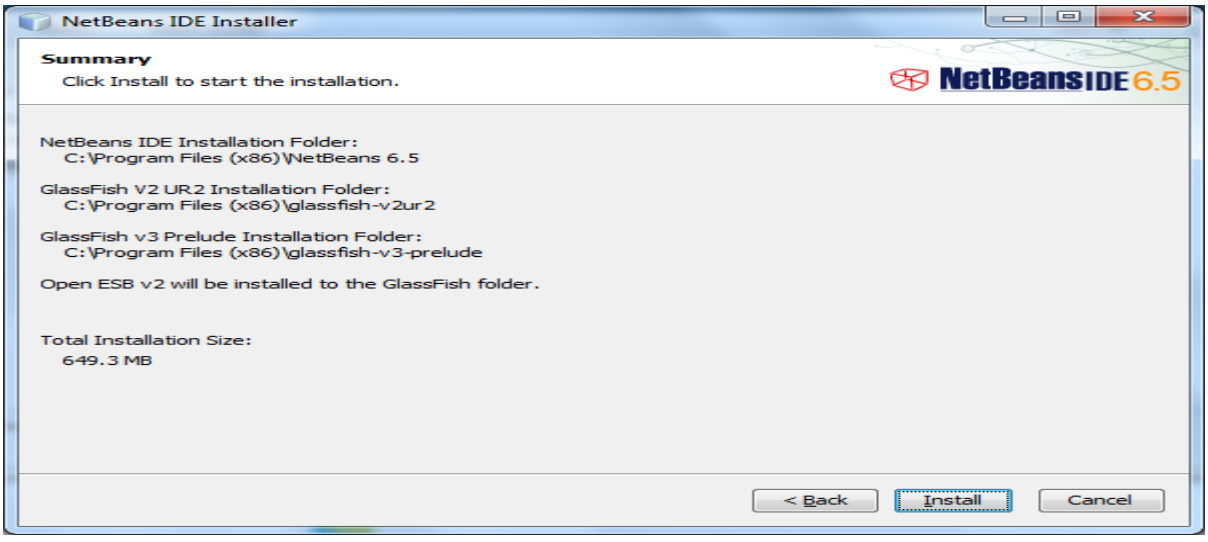
10. Set the **Password, User Name & Ports** for the Network Connectivity, or we can use this **UserName and the Password** for the Connecting the Front-End to the Back-End.



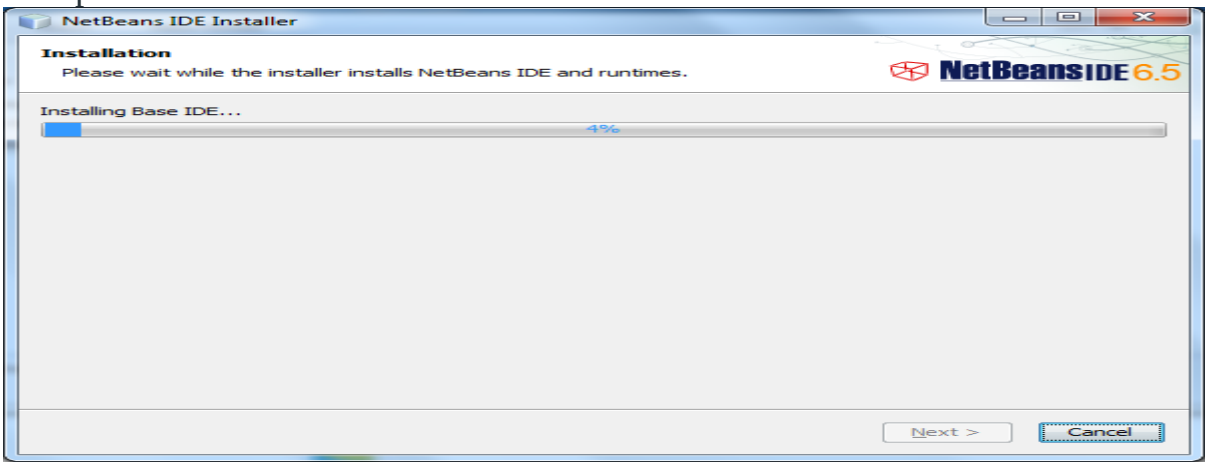
11. Click on the "**Next**" button.



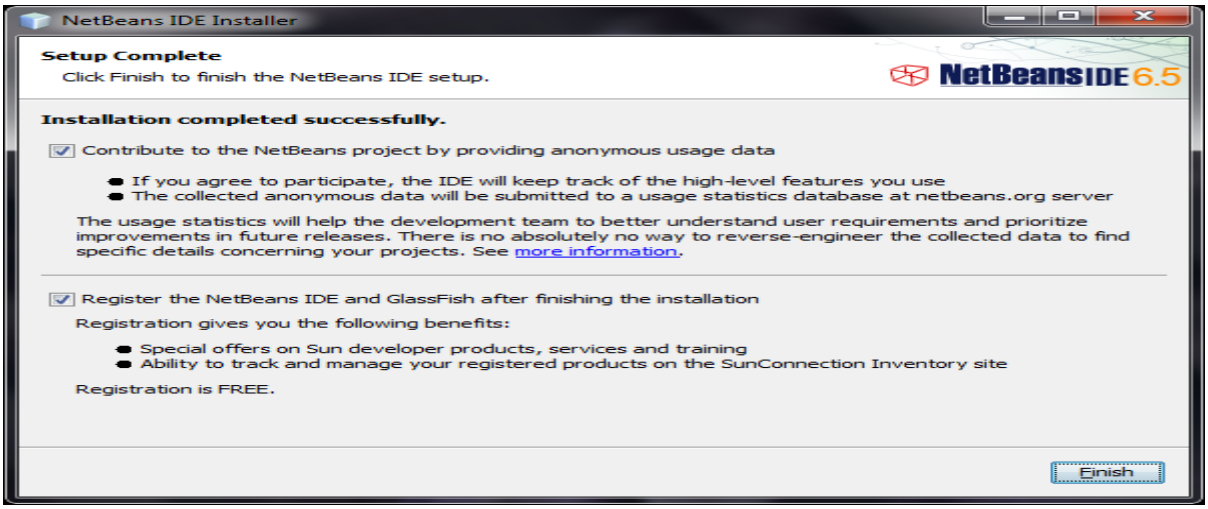
12. Click on the "**Install**" button.



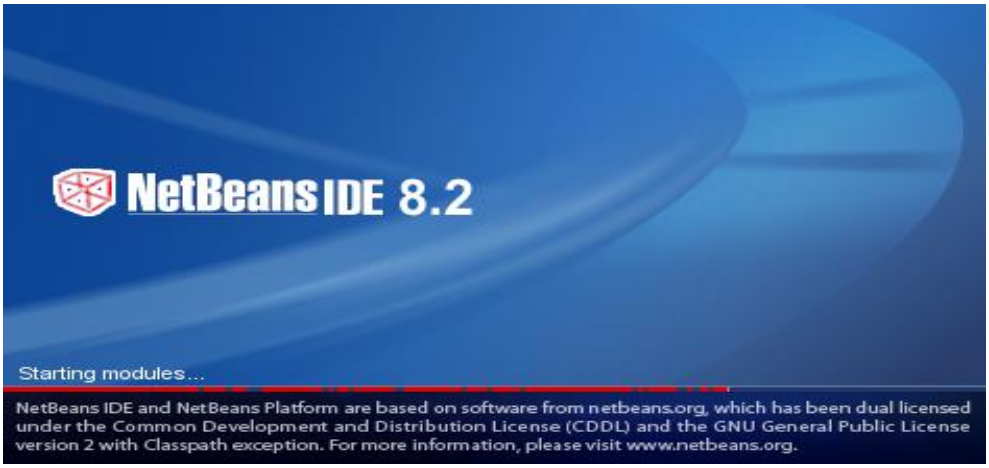
13. Wait for the while till the time the setup is properly Installed into the Computer



14. After complication of the setup you can click on the "**Finish"** button or you can also register the Software, for Further Assistance because it is a Free Software.



15. Now you can start the NetBeans for further use



Practical – 2

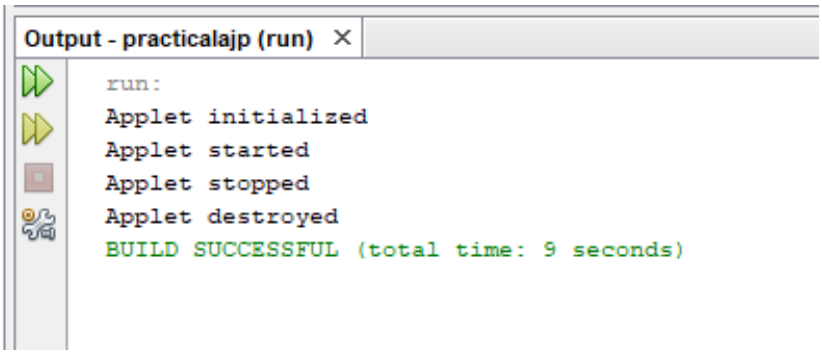
AIM: WAP to demonstrate life cycle of Applet.

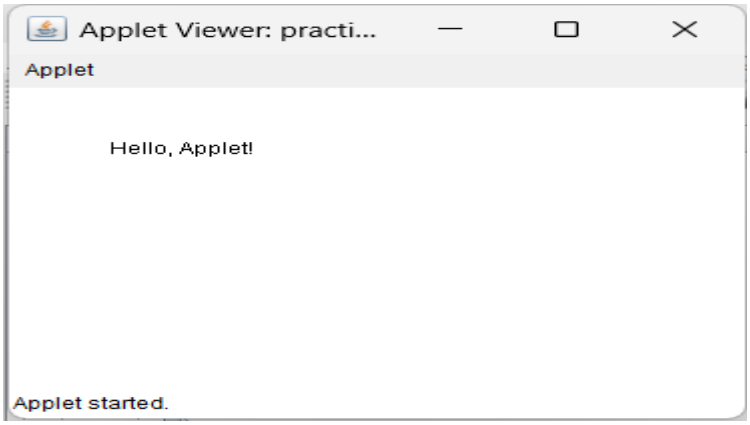
Code:

```
package practicalajp;


import java.applet.Applet;
import java.awt.Graphics;
/**
 * Practical 2: WAP to demonstrate life cycle of Applet.
 *@author Aman Patel
 *Name:Patel Aman Sajid bhai
 *Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */
public class AJP_pr2_QC2 extends Applet {
@Override
    public void init() {
        System.out.println("Applet initialized");
    }
     @Override
    public void start () {
        System.out.println("Applet started");
    }
    @Override
    public void stop () {
        System.out.println("Applet stopped");
```

```
    }
    @Override
    public void destroy () {
        System.out.println("Applet destroyed");
    }
    @Override
    public void paint (Graphics g) {
        g.drawString("Hello, Applet!", 50, 50);
    }
}
```

OUTPUT:





Practical-3

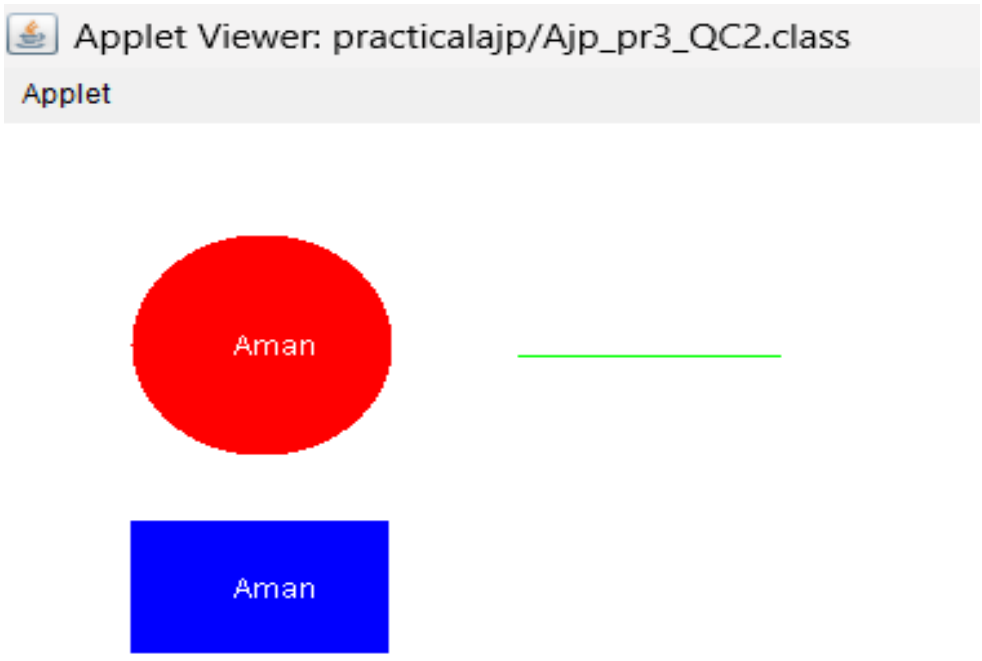
Aim: WAP to draw the line, rectangle, oval using graphic class.

Code:

```
package practicalajp;
import java. applet.Applet;
import java.awt. Graphics;
import java.awt. Color;

/**
* Practical 3: WAP to draw the line, rectangle, oval using graphic class.
*@author Aman Patel
 *Name: Patel Aman Sajid bhai
 *Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */
public class Ajp_pr3_QC2 extends Applet
{
   @Override
   public void paint (Graphics g) {

       //For Oval
       g.setColor(Color.red);
       g.fillOval(50, 50, 100, 100);
       g.setColor(Color.white);
       g.drawString("Aman", 90, 105);
       //For Rectangle
```

```
        g.setColor(Color.blue);
        g.fillRect(50, 180, 100, 60);
        g.setColor(Color.white);
        g.drawString("Aman", 90, 215);
       //For Line
        g.setColor(Color.green);
        g.drawLine(300, 105, 200, 105);
    }
}
```

Output:



Practical-4

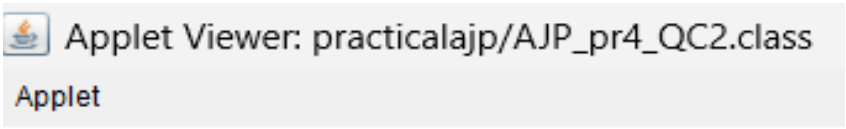
Aim: WAP to draw the smiley using graphics class.

Code:

```
package practicalajp;


import java.applet.Applet;
import java.awt.Color;
import java.awt.Graphics;


/**
 * Practical 4: WAP to draw the smiley using graphics class.
 *@author Aman Patel
 *Name:Patel Aman Sajid bhai
 *Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */
public class AJP_pr4_QC2 extends Applet
{
    @Override
    public void init() {
        // TODO start asynchronous download of heavy resources
    }
    @Override
    public void paint(Graphics g) {
g.setColor(Color.yellow);
        g.fillOval(50,50,200,200);
```

```
        g.setColor(Color.RED);
        g.fillOval(100,100,40,40);
        g.fillOval(190,100,40,40);

        g.setColor(Color.red);
        g.fillArc(100,120,130,80,0,-180);
    }
}
```

Output:



Practical – 5

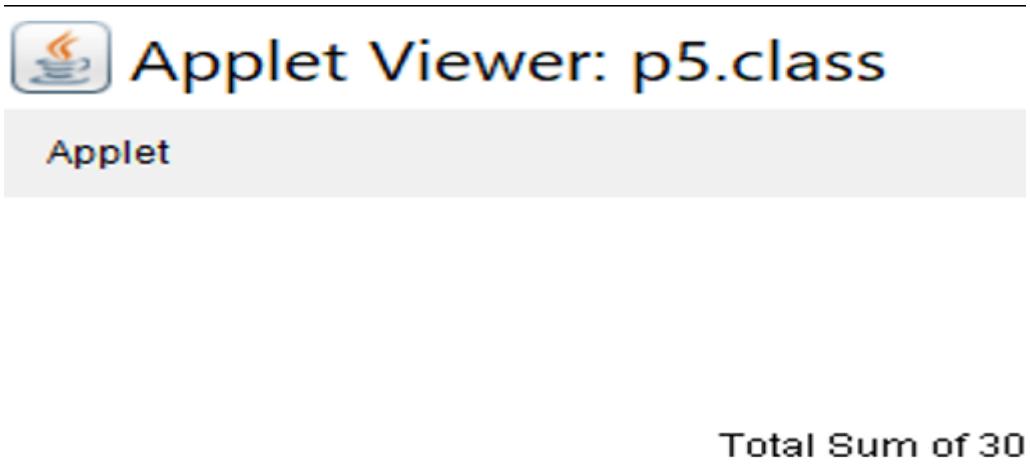
Aim: WAP to perform addition of two numbers using PARAM keyword.

Code :

```
/*
<applet code="p5.class"
width=1920 height="920">
<param name="a" value="10">
<param name="b" value="20">
</applet>
*/

import java.applet.Applet;
import java.awt.Graphics;

/**
 * Practical 5: WAP to perform addition of two numbers using PARAM keyword.
*@author Aman Patel
 *Name: Patel Aman Sajid bhai
 *Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */

public class p5 extends Applet {
        int a, b, c;
```

```
        String x, y;

        public void init() {

        }

        public void paint(Graphics g) {
                x = getParameter("a");
                y = getParameter("b");
                a = Integer.parseInt(x);
                b = Integer.parseInt(y);
                c = a + b;
                g.drawString("Practical 5 AJP QC1", 50, 50);
                g.drawString("Ans is " + c, 50, 100);
        }
}
```

Output :



Practical-6

Aim: Draw ten circles and rectangle in a horizontal and fill with random color in the center of the applet.

Code:

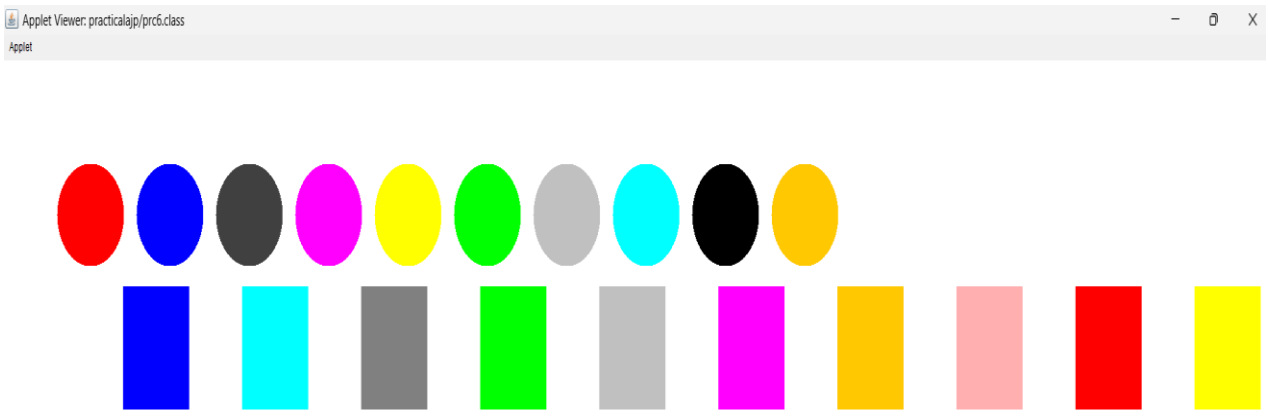
```
package practicalajp;


import java.applet.Applet;
import java.awt.Color;
import java.awt.Graphics;


/**
 * Practical 6: Draw ten circles  and rectangle in a horizontal and fill with random color in the center of the applet.
 *@author Aman Patel
 *Name:Patel Aman Sajid bhai
 *Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */
public class prc6 extends Applet {
/**
     * Initialization method that will be called after the applet is loaded into
     * the browser.
     */
    @Override
    public void init() {
        // TODO start asynchronous download of heavy resources
    }
```

```
@Override
 public void paint(Graphics g)
 {
    g.setColor(Color.red);
    g.fillOval(80, 100, 100, 100);

    g.setColor(Color.blue);
    g.fillOval(200, 100, 100, 100);

    g.setColor(Color.DARK_GRAY);
    g.fillOval(320, 100, 100, 100);

    g.setColor(Color.MAGENTA);
    g.fillOval(440, 100, 100, 100);

    g.setColor(Color.YELLOW);
    g.fillOval(560, 100, 100, 100);

    g.setColor(Color.green);
    g.fillOval(680, 100, 100, 100);

    g.setColor(Color.lightGray);
    g.fillOval(800, 100, 100, 100);

    g.setColor(Color.cyan);
    g.fillOval(920, 100, 100, 100);
```

```
g.setColor(Color.black);
g.fillOval(1040, 100, 100, 100);

g.setColor(Color.orange);
g.fillOval(1160, 100, 100, 100);

g.setColor(Color.BLUE);
g.fillRect(180, 220, 100, 120);

g.setColor(Color.CYAN);
g.fillRect(360, 220, 100, 120);

g.setColor(Color.GRAY);
g.fillRect(540, 220, 100, 120);

g.setColor(Color.GREEN);
g.fillRect(720, 220, 100, 120);

g.setColor(Color.LIGHT_GRAY);
g.fillRect(900, 220, 100, 120);

g.setColor(Color.MAGENTA);
g.fillRect(1080, 220, 100, 120);

g.setColor(Color.ORANGE);
g.fillRect(1260, 220, 100, 120);
```

```
        g.setColor(Color.PINK);
        g.fillRect(1440, 220, 100, 120);

        g.setColor(Color.RED);
        g.fillRect(1620, 220, 100, 120);

        g.setColor(Color.YELLOW);
        g.fillRect(1800, 220, 100, 120);

    }
}
```

Output:



Practical-7

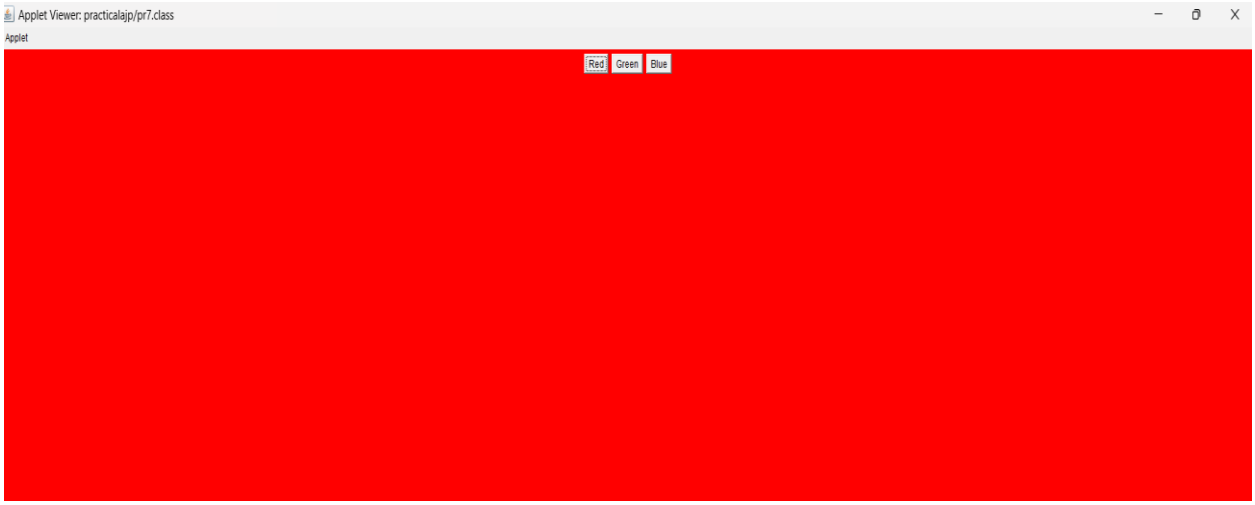
Aim: WAP to change background color of Applet using three button RED, GREEN, BLUE. If user click on red background should be red and so on.

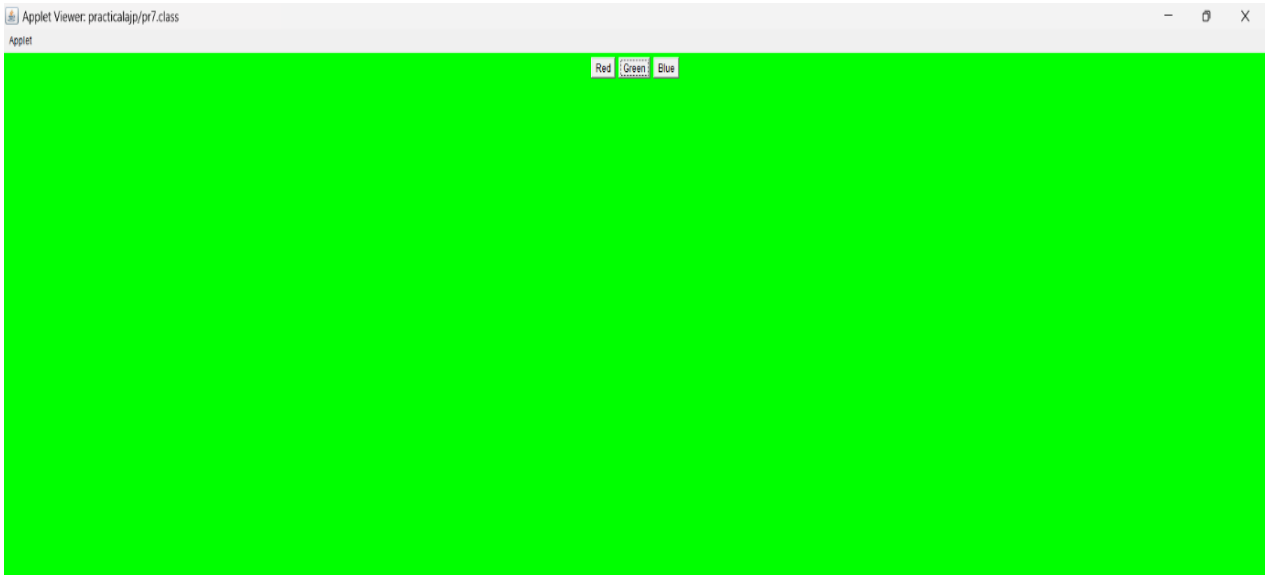
Code:

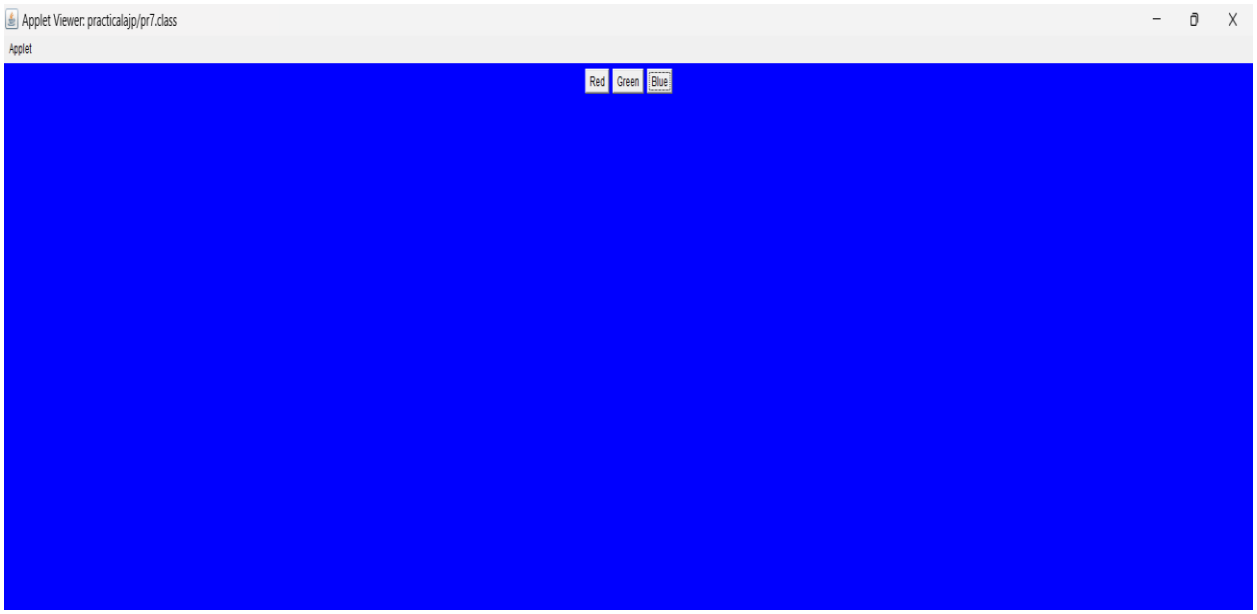
```
package practicalajp;


import java.applet.Applet;
import java.awt.Button;
import java.awt.Color;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;


/**
 * Practical 7: WAP to change background color of Applet using three button RED, GREEN, BLUE. If user click on red background should be red and so on.
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
 *Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */



public class pr7 extends Applet implements ActionListener {


    Button br;
    Button bg;
```

```
    Button bb;

    @Override
    public void init() {
        br = new Button("Red");
        bg = new Button("Green");
        bb = new Button("Blue");
        add(br);
        add(bg);
        add(bb);
        br.addActionListener(this);
        bg.addActionListener(this);
        bb.addActionListener(this);
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == br) {
            setBackground(Color.RED);
        } else if (e.getSource() == bg) {
            setBackground(Color.GREEN);
        } else if (e.getSource() == bb) {
            setBackground(Color.BLUE);
        }
    }
}
```

Output:


Applet Viewer: practicalajp/pr7.class
Applet
Red
Green
Blue


Applet Viewer: practicalajp/pr7.class
Applet
Red
Green
Blue


Applet Viewer: practicalajp/pr7.class
Applet
Red
Green
Blue

Practical – 8

AIM: WAP to prepare registration form of students in which add following information like student no, name, address, phone no, gender, hobbies, city, college etc. and clock on submit button all information should be display on Applet.

Code:

```
package practicalajp;


import java.applet.Applet;
import java.awt.Button;
import java.awt.Checkbox;
import java.awt.CheckboxGroup;
import java.awt.Choice;
import java.awt.Graphics;
import java.awt.Label;
import java.awt.TextField;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;


/**
 * Practical 8 : WAP to prepare registration form of students in which add following information like student no, name, address, phone no, gender, hobbies, city, college etc. and clock on submit button all information should be display on Applet.
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
```

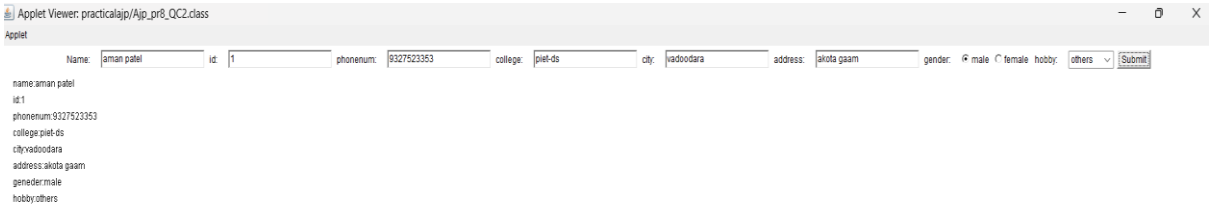
```
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */
public class Ajp_pr8_QC2 extends Applet implements ActionListener {

    String name ="",
           id ="",
           phonenum="",
           college="",
           city="",
           address="",
           geneder="",
           hobby="";

    Label L1 = new Label("Name:");
    TextField tf1 = new TextField(20);
    Label L2 = new Label ("id:");
    TextField tf2 = new TextField(20);
    Label L3 = new Label("phonenum:");
    TextField tf3 = new TextField(20);
    Label L4 = new Label("college:");
    TextField tf4 = new TextField(20);
    Label L5 = new Label("city:");
    TextField tf5 = new TextField(20);
    Label L6 = new Label("address:");
    TextField tf6 = new TextField(20);
```

```
    Button b = new Button("Submit");
    Label L7 = new Label("gender:");
    CheckboxGroup g = new CheckboxGroup();
    Checkbox m = new Checkbox("male",g,true);
    Checkbox f = new Checkbox("female",g,false);
    Label L8 = new Label("hobby:");
    Choice c = new  Choice();

    @Override
public void init() {
        add(L1);
        add(tf1);
        add(L2);
        add(tf2);
        add(L3);
        add(tf3);
        add(L4);
        add(tf4);
        add(L5);
        add(tf5);
        add(L6);
        add(tf6);
        add(L7);
        add(m);
        add(f);
        add(L8);
        c.add("singing");
```

```
    c.add("dancing");
    c.add("playing");
    c.add("reading");
    c.add("gaming");
    c.add("others");
    add(c);
    add(b);
    b.addActionListener(this);
  }

  public void paint(Graphics g)
  {
     g.drawString("name:"+name, 20, 50);
     g.drawString("id:"+id, 20, 70);
     g.drawString("phonenum:"+phonenum, 20, 90);
     g.drawString("college:"+college, 20, 110);
     g.drawString("city:"+city, 20, 130);
     g.drawString("address:"+address, 20, 150);
     g.drawString("geneder:"+geneder, 20, 170);
     g.drawString("hobby:"+hobby, 20, 190);
  }

  @Override
  public void actionPerformed(ActionEvent e) {
    name=tf1.getText();
    id=tf2.getText();
    phonenum=tf3.getText();
```

```
        college=tf4.getText();
        city=tf5.getText();
        address=tf6.getText();
        geneder=g.getSelectedCheckbox().getLabel();
        hobby = c.getSelectedItem();

        repaint();
    }
}
```

Output:



Practical : 9

AIM :- write a Program using awt to create a menu bar where menu bar contains menu items such as file, view and create a submenu under the file menu: new and open.

Solution :-

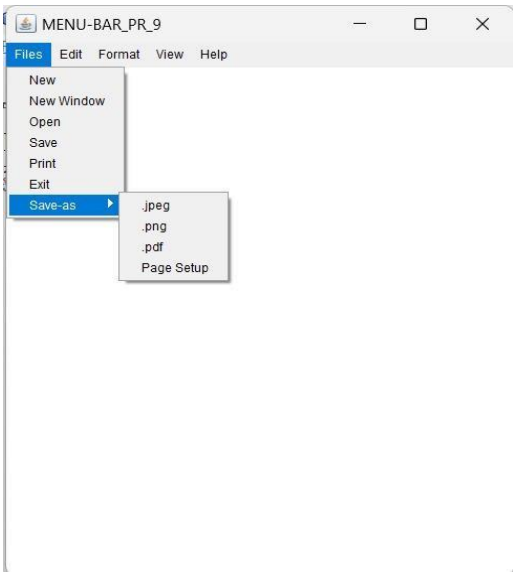
```
import java.awt.Frame;
import java.awt.Menu;
import java.awt.MenuBar;
 import java.awt.MenuItem;
import java.awt.event.*;
/**

*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */
public class MENU-BAR_PR_9
extends Frame
{
public static void main(String[]
args)
{
```

```
//create object for frame
Frame f=new Frame("Frame");
//Create object for MenuBar
MenuBar mb=new MenuBar();
f.setMenuBar(mb);
//Create object For Menu add diffrent menu of notepad
Menu m=new Menu("File");
Menu m1=new Menu("Edit");
//Menu m2=new Menu("Format");
Menu m3=new Menu("View");
//Menu m4=new Menu("Help");
mb.add(m);
mb.add(m1);
mb.add(m3);
MenuItem mn=new MenuItem("New Tab");
MenuItem mnw=new MenuItem("New Window");
MenuItem mo=new MenuItem("Open");
MenuItem ms=new MenuItem("Save");
MenuItem msa=new MenuItem("Save as");
MenuItem mp=new MenuItem("Print");
MenuItem me=new MenuItem("Exit");
m.add(mn);
m.add(mnw);
m.add(mo);
m.add(ms);
m.add(msa);
```

```
m.add(mp);
m.add(me);
MenuItem mu=new MenuItem("undo");
MenuItem mc=new MenuItem("cut");
MenuItem mcp=new MenuItem("copy");
MenuItem mpp=new MenuItem("paste");
MenuItem md=new MenuItem("delete");
MenuItem mswb=new MenuItem("search with bing");
MenuItem mf=new MenuItem("find");
MenuItem mfn=new MenuItem("find next");
MenuItem mfp=new MenuItem("find previous");
MenuItem mr=new MenuItem("Replace");
MenuItem mgt=new MenuItem("go tob");
MenuItem mtd=new
MenuItem("Time/Date"); MenuItem
mfnt=new MenuItem("Font"); m1.add(mu);
m1.add(mc); m1.add(mcp); m1.add(mpp);
m1.add(md); m1.add(mswb); m1.add(mf);
m1.add(mfn);

m1.add(mfp);
m1.add(mr);
m1.add(mgt);
m1.add(mtd);
m1.add(mfnt);
```

```
MenuItem mz=new MenuItem("Zoom");
MenuItem msb=new MenuItem("Status Bar");
MenuItem mww=new MenuItem("Word
wrap"); m3.add(mz); m3.add(msb);
m3.add(mww);
// Add a window listener to close the application
f.addWindowListener(new WindowAdapter()
{
public void windowClosing(WindowEvent we)
{
System.exit(0);
}
});
f.setTitle("Notepad");
f.setVisible(true);
}
}
```

Output:-



Practical : 10

AIM :- Write to demonstrate flow layout, card layout, Border layout grid layout. Solution:-

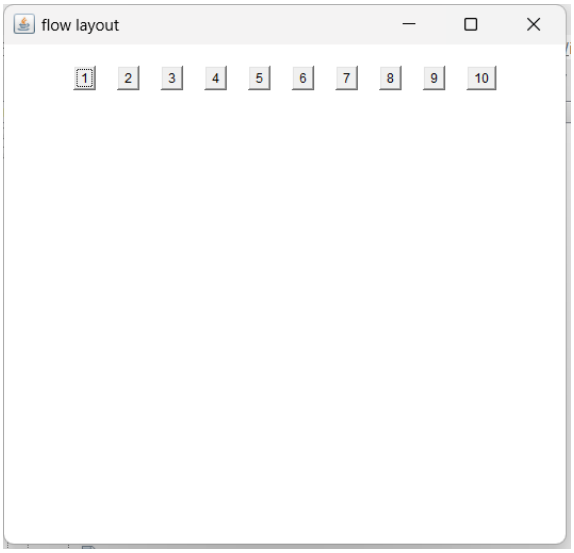
//FlowLayout

```
/*
 * 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.
 */
package practicalajp;

import java.applet.Applet;
import java.awt.FlowLayout;
import java.awt.Frame;
import java.awt.Button;

/**
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */


public class prc10 extends Applet {
    public static void main (String args[])
```

```
    {
        Frame f1=new Frame("flow layout");
        f1.setSize(300,300);
        f1.setLayout(new FlowLayout(FlowLayout.CENTER,20,20));
        f1.add(new Button("1"));
         f1.add(new Button ("2"));
        f1.add(new Button("3"));
        f1.add(new Button ("4"));
        f1.add(new Button("5"));
        f1.add(new Button ("6"));
        f1.add(new Button("7"));
        f1.add(new Button ("8"));
        f1.add(new Button("9"));
        f1.add(new Button ("10"));
        f1.setVisible(true);
}

    /**
     * Initialization method that will be called after the applet is loaded into
     * the browser.
     */
    @Override
    public void init() {
        // TODO start asynchronous download of heavy resources
    }

    // TODO overwrite start(), stop() and destroy() methods
```

}

Output:-



// Border layout :-

```
package practicalajp;
import java.applet.Applet;
import java.awt.BorderLayout;
import java.awt.Frame;
import java.awt.Button;

public class prc10_2 extends Applet {
   public static void main (String args[])
   {
      Frame f1=new Frame("border layout");

      f1.setSize(300,300);
      Button b1,b2,b3,b4,b5;
      b1 =new Button ("button 1");
```

```
        b2 =new Button ("button 2");
        b3 =new Button ("button 3");
        b4 =new Button ("button 4");
        b5 =new Button ("button 5");

           //add to Button in Frame

        f1.add(b1,BorderLayout.NORTH);
        f1.add(b2,BorderLayout.SOUTH);
        f1.add(b3,BorderLayout.EAST);
        f1.add(b4,BorderLayout.WEST);
        f1.add(b5,BorderLayout.CENTER);
        f1.setVisible(true);
          }

    /**
     * Initialization method that will be called after the applet is loaded into
     * the browser.
     */
    @Override
    public void init() {
        // TODO start asynchronous download of heavy resources
    }

    // TODO overwrite start(), stop() and destroy() methods
}
```

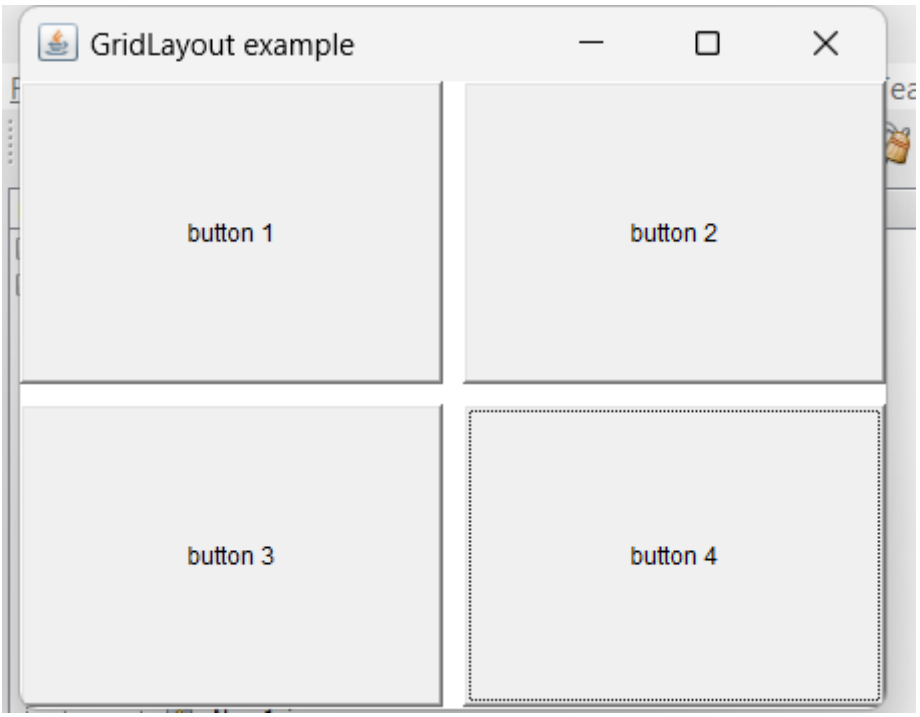
Output:-

Grid Lay:-

```
import java.applet.Applet;
import java.awt.GridLayout;
import java.awt.Button;
import java.awt.Frame;

public class prc10_3 extends Applet{

   public static void  main(String args[])
   {
      //create frame
      Frame f1=new Frame("GridLayout example");
      f1.setSize(300,300);

      // create component
```

```
        Button b1=new Button("button 1");
        Button b2=new Button("button 2");
        Button b3=new Button("button 3");
        Button b4=new Button("button 4");

        //set layout
        f1.setLayout(new GridLayout(2,2,10,10));

        //add compnent in frame
        f1.add(b1);
        f1.add(b2);
        f1.add(b3);
        f1.add(b4);
        f1.setVisible(true);

    }
}
```

Output:-



Practical:11

Aim: WAP to create a simple calculator with the use of grid layout.

Solution:

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

/**
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */

public class prc11 {
    Frame frame;
    TextField textField;
    Button[] buttons = new Button[16];
    String[] buttonLabels = {"7", "8", "9", "/",
                             "4", "5", "6", "*",
                             "1", "2", "3", "-",
                             "0", ".", "=", "+"};

    public prc11() {
        frame = new Frame("Calculator");
        frame.setLayout(new BorderLayout());
```

```
        // Create text field for displaying numbers
        textField = new TextField();
        frame.add(textField, BorderLayout.NORTH);

        // Create panel for buttons
        Panel buttonPanel = new Panel();
        buttonPanel.setLayout(new GridLayout(4, 4));

        // Create buttons
        for (int i = 0; i < 16; i++) {
            buttons[i] = new Button(buttonLabels[i]);
            buttonPanel.add(buttons[i]);
            buttons[i].addActionListener(new ButtonListener());
        }

        frame.add(buttonPanel, BorderLayout.CENTER);

        frame.pack();
        frame.setVisible(true);
    }

    // Inner class for button listener
    class ButtonListener implements ActionListener {
        public void actionPerformed(ActionEvent e) {
            String command = e.getActionCommand();
            if (command.equals("=")) {
```

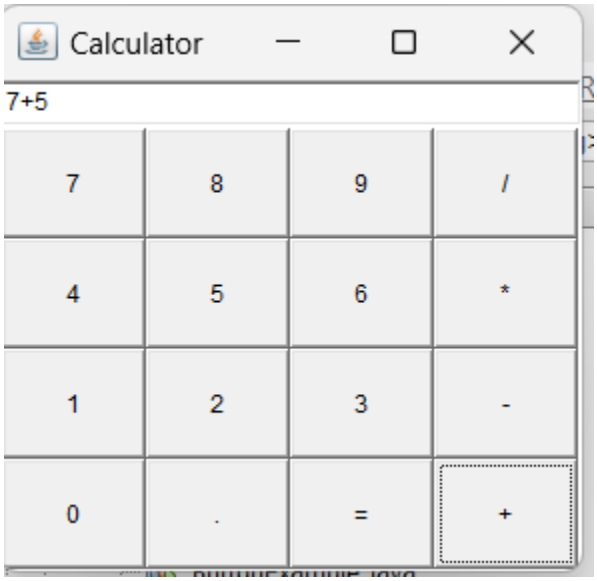
```
            try {
                double result = evaluate(textField.getText());
                textField.setText(String.valueOf(result));
            } catch (Exception ex) {
                textField.setText("Error");
            }
        } else {
            textField.setText(textField.getText() + command);
        }
    }
}

// Method to evaluate mathematical expression
double evaluate(String expression) throws Exception {
    return new Object() {
        int pos = -1, ch;

        void nextChar() {
            ch = (++pos < expression.length()) ? expression.charAt(pos) : -1;
        }

        boolean eat(int charToEat) {
            while (ch == ' ') nextChar();
            if (ch == charToEat) {
                nextChar();
                return true;
            }
```

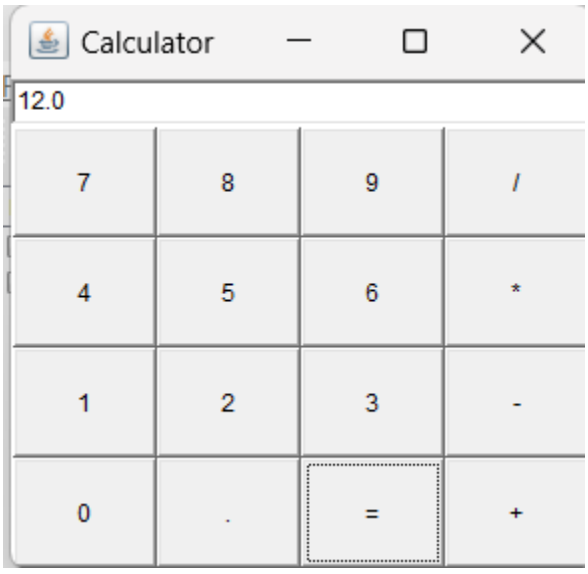
```
        return false;
    }

    double parse() throws Exception {
        nextChar();
        double x = parseExpression();
        if (pos < expression.length()) throw new Exception("Unexpected: " +
(char)ch);
        return x;
    }

    double parseExpression() throws Exception {
        double x = parseTerm();
        for (;;) {
            if      (eat('+')) x += parseTerm(); // addition
            else if (eat('-')) x -= parseTerm(); // subtraction
            else return x;
        }
    }

    double parseTerm() throws Exception {
        double x = parseFactor();
        for (;;) {
            if      (eat('*')) x *= parseFactor(); // multiplication
            else if (eat('/')) x /= parseFactor(); // division
            else return x;
        }
    }
```

```
        double parseFactor() throws Exception {
            if (eat('+')) return parseFactor(); // unary plus
            if (eat('-')) return -parseFactor(); // unary minus

            double x;
            int startPos = this.pos;
            if (eat('(')) { // parentheses
                x = parseExpression();
                eat(')');
            } else if ((ch >= '0' && ch <= '9') || ch == '.') { // numbers
                while ((ch >= '0' && ch <= '9') || ch == '.') nextChar();
                x = Double.parseDouble(expression.substring(startPos, this.pos));
            } else if (ch >= 'a' && ch <= 'z') { // functions
                while (ch >= 'a' && ch <= 'z') nextChar();
                String func = expression.substring(startPos, this.pos);
                x = parseFactor();
                // if (func.equals("sqrt")) x = Math.sqrt(x);
                // else throw new Exception("Unknown function: " + func);
            } else {
                throw new Exception("Unexpected: " + (char)ch);
            }

            return x;
        }
    }.parse();
}
```

```
    public static void main(String[] args) {
        new prc11();
    }
}
```

Output:-





Practical:12

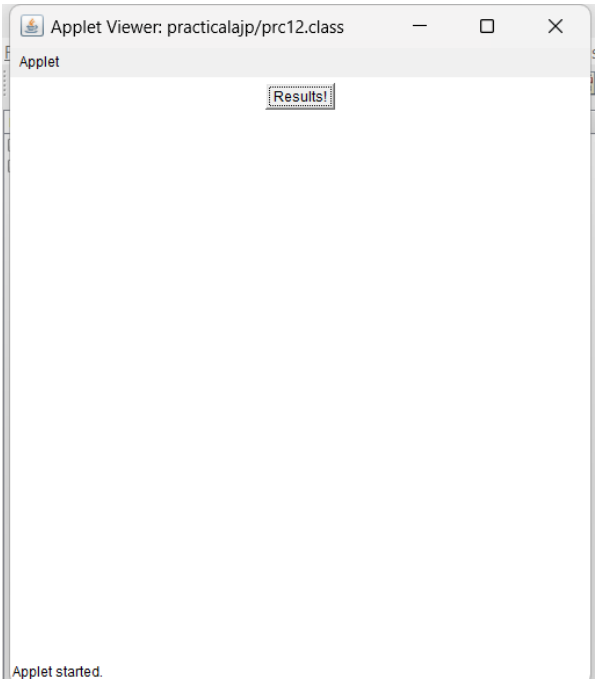
Aim: Write a program to create a dialog box.

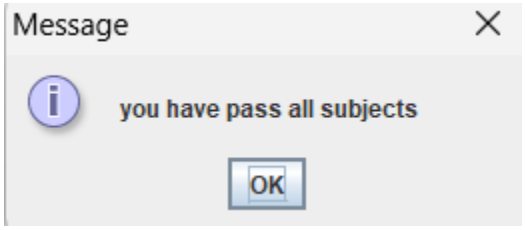
Solution:

```
import java.applet.Applet;
import java.awt.Button;
import java.awt.Rectangle;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.JOptionPane;

/**
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */

public class prc12 extends Applet implements
ActionListener {


    @Override
    public void init() {

      setSize(500,500);
      Button b1=new Button("Results!");
      b1.setBounds(new Rectangle(10,10,20,30));
      b1.addActionListener(this);
      add(b1,null);
    }
```

```
    @Override
    public void actionPerformed(ActionEvent e) {

        String s =e.getActionCommand();
        if(s.equals("Results!"))
            JOptionPane.showMessageDialog(null,
new String("you have pass all subjects"));
    }
}
```

Output:-





Practical:12.2

Aim: Write a program to create a dialog box.

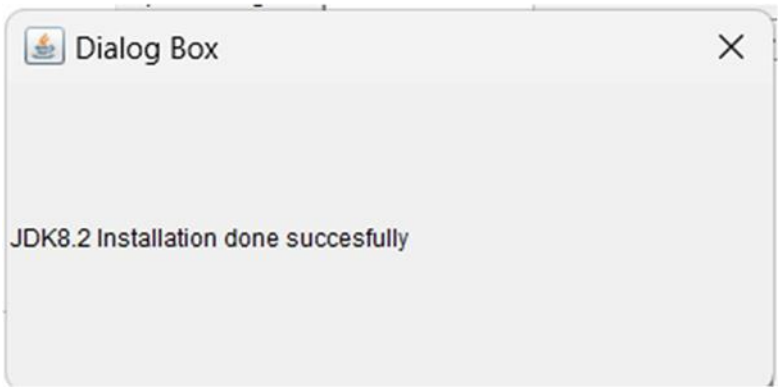
Solution:

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


/**
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */


public class Pr12_2 extends WindowAdapter implements
ActionListener
{
   Frame frame;
   Button button1;
   Label label1;
   Dialog d1;


  Pr12_2()
    {
      frame=new Frame("Dialog Box!");
      button1=new Button("Open Dialog Box!");
      label1=new Label("Click on the button to
open a dialog box");
      frame.add(button1);
      frame.add(label1);
```

```
      button1.addActionListener(this);
      frame.setLayout((LayoutManager) new
Flowlayout());
      frame.setSize(330,250);
      frame.setVisible(true);
   }


 @Override
   public void actionPerformed(ActionEvent e)
   {
      if(e.getActionCommand().equals("Open
dailog box"))
      {
         d1=new Dialog(frame,"dialog box",true);
         Label label=new Label("JDK8.2 Installation done
successfully");
         d1.add(label);
         d1.addWindowListener(this);
         d1.pack();
         d1.setLocationRelativeTo(frame);
         d1.setSize(400,200);
         d1.setVisible(true);
      }
   }


 @Override
   public void windowClosing(WindowEvent
we)
{
   d1.setVisible(false);
}

public static void main(String[] args)
{
      Pr12_2 pr12_2 = new Pr12_2();
}
}
```

Output:


Dialog Box
Click on the button to open a Dialog box
Open Dialog Box


Dialog Box
JDK8.2 Installation done succesfully

Practical:14

Aim: Create a chat application on either TP or UDP protocol.

Solution:

TCP Server

```
import java.io.*;
import java.net.*;
/**
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */

public class TCPServer
{
   public static void main(String[] args) throws IOException {

// Create a server socket on port 6789
        ServerSocket serverSocket = new ServerSocket(6789);
        System.out.println("Server started and listening on port 6789");

// Accept a client connection
        Socket clientSocket = serverSocket.accept();
        System.out.println("Accepted client connection from: "
+clientSocket.getRemoteSocketAddress());

// Get input and output streams from the client socket
```

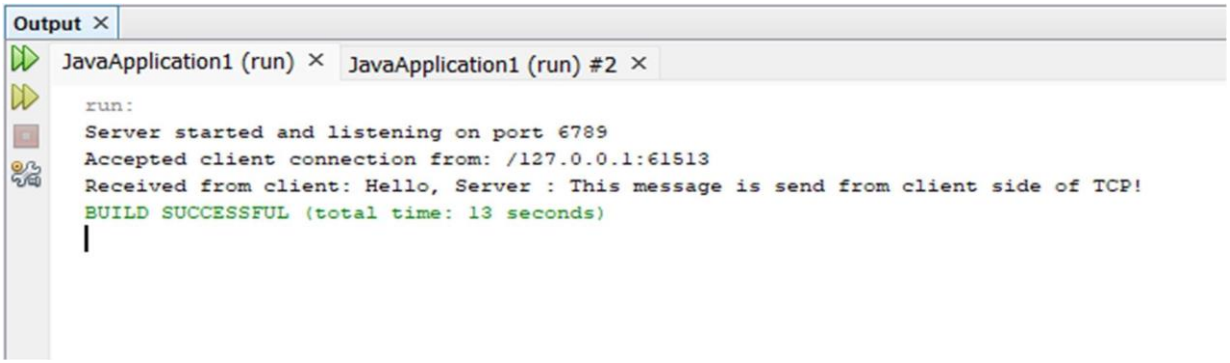
```
        InputStream inputStream = clientSocket.getInputStream();
        OutputStream outputStream = clientSocket.getOutputStream();

// Read data from the input stream
        BufferedReader reader = new BufferedReader(new
InputStreamReader(inputStream));
        String line = reader.readLine();
        System.out.println("Received from client: " + line);

// Write data to the output stream
        PrintWriter writer = new PrintWriter(outputStream, true);
        writer.println("Hello, client : This message is send from server side of
TCP!!");

// Close the client socket
        clientSocket.close();
    }
}
```

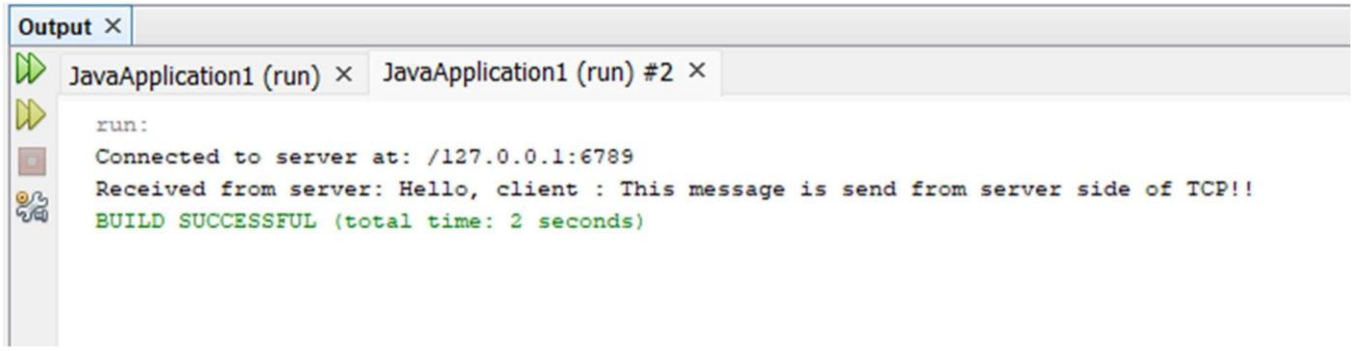
TCPClient

```
import java.io.*;
import java.net.*;
/**
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */
```

```
public class TCPClient
{
   public static void main(String[] args) throws IOException {

// Create a client socket and connect to the server on IP address 127.0.0.1 and port 6789
        Socket clientSocket = new Socket("127.0.0.1", 6789);
        System.out.println("Connected to server at: " + clientSocket.getRemoteSocketAddress());

// Get input and output streams from the client socket
        InputStream inputStream = clientSocket.getInputStream();
        OutputStream outputStream = clientSocket.getOutputStream();



// Write data to the output stream
        PrintWriter writer = new PrintWriter(outputStream, true);
        writer.println("Hello, Server : This message is send from client side of TCP!");

// Read data from the input stream
        BufferedReader reader = new BufferedReader(new InputStreamReader(inputStream));
        String line = reader.readLine();
        System.out.println("Received from server: " + line);

// Close the client socket
        clientSocket.close();
    }
}
```

Output:-


Output
JavaApplication1 (run)
JavaApplication1 (run) #2
run:
Server started and listening on port 6789
Accepted client connection from: /127.0.0.1:61513
Received from client: Hello, Server : This message is send from client side of TCP!
BUILD SUCCESSFUL (total time: 13 seconds)


Output
JavaApplication1 (run)
JavaApplication1 (run) #2
run:
Connected to server at: /127.0.0.1:6789
Received from server: Hello, client : This message is send from server side of TCP!!
BUILD SUCCESSFUL (total time: 2 seconds)

Practical:15

Aim: Implement TCP Server for transferring files using Socket and Server Socket of each string.

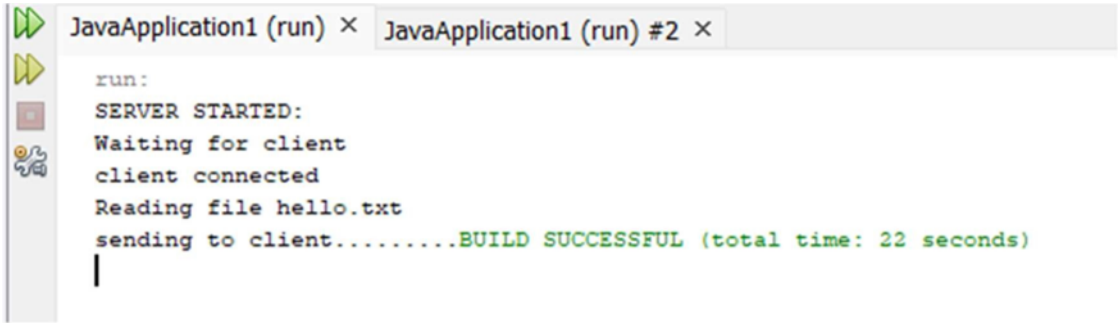
Solution:

Server

```
import java.io.*;
import java.net.*;

/**

*@author Aman Patel

 *Name:Patel Aman Sajid bhai

*Enrollment No:2203396160199

 *Division:5QC

 * Batch:2

 */

public class Server
{
 public static void main(String ar[]) throws
Exception
 {
   ServerSocket s1=new ServerSocket(12345);
   System.out.println("SERVER
STARTED:\nWaiting for client");

   Socket s=s1.accept();
   System.out.println("client connected");
   PrintWriter p=new
PrintWriter(s.getOutputStream(),true);
   BufferedReader in=new BufferedReader(new
InputStreamReader(s.getInputStream()));

   String st=in.readLine();
   FileReader fr=new FileReader(st);
```

```
    BufferedReader br=new BufferedReader(fr);
    String line=new String();

   System.out.println("Reading file "+st);
   System.out.print("sending to client");

   for(int i=1;i<10;i++)
  {
     System.out.print(".");
     Thread.sleep(100);
   }
 while((line=br.readLine())!=null)
 {
    p.println(line);
  }
 p.flush();
 s.close();
}
}
```

Output:-

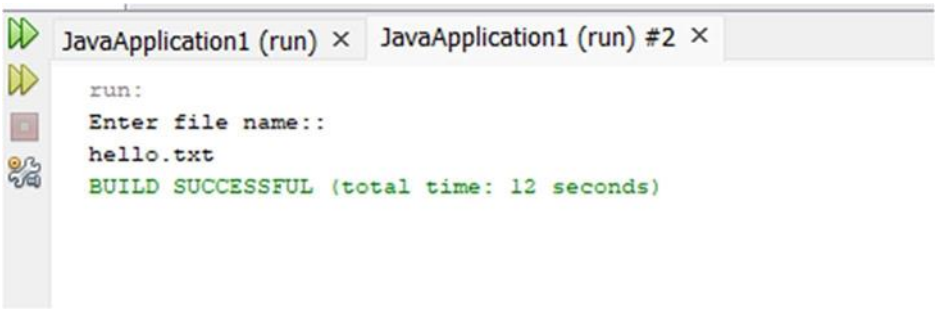


Client

```
Import java.io.*;
Import java.net.*;
class Client
{
public static void main(String ar[])
        throws Exception
{
```

```
    Socket s=new Socket("localhost",12345);
    DataOutputStream dos=new DataOutputStream(s.getOutputStream());
    BufferedReader in=new BufferedReader(new
InputStreamReader(s.getInputStream()));
    BufferedReader ink=new BufferedReader(new
InputStreamReader(System.in));


    System.out.println("Enter file name::");
    String s1=ink.readLine();
    dos.writeBytes(s1+"\n");
        String sr;

        while((sr=in.readLine())!=null)
        {
                System.out.println(sr);
        }
        s.close();
}
}
```

Output:-



Practical 16

Aim: Develop a database application that uses ODBC JDBC driver.

```
import java.sql.*;
/**
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */

 public class Prc16 {
public static void main(String args[]) {
try{Class.forName("com.mysql.cj.jdbc.
Driver");

 Connection con =
DriverManager.getConnection("jdbc:my
sql://localhost:3306/ajp", "root", "");

          Statement stmt = con.createStatement();
          ResultSet rs = stmt.executeQuery("select * from studentdata");
          while (rs.next())
             System.out.println(rs.getInt(1) + " " + rs.getString(2));
// This is going to print my data that is fetch from database.
          con.close();
       } catch (Exception e) {
          System.out.println(e);
```

```
    } }
}
```

OUTPUT:-

enroll	name
1	bbb
2	aaa

```
run:
1 bbb
2 aaa
BUILD SUCCESSFUL (total time: 0 seconds)
```

Practical 17

Aim: Develop a database application that insert, delete, update data from database using type 4 JDBC driver.

```
import java.sql.*;
import java.util.logging.Level;
import java.until.logging.*;
/**
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */
 public class Prac17 {
public static void main(String[] args)
{
 try{
 try {
//step1 load the driver class
Class.forName("com.mysql.cj.jdbc.Driver");
}
catch (ClassNotFoundException ex)
 {
Logger.getLogger(Prac17.class.getName()).log(Level.SEVERE, null, ex);
```

```
}
try
{
//step2 create the connection object driver url uname pass
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/sample","root" ,
""))
 {
System.out.println("Connection Establish Successfully");
//step3 create the statement object
Statement stmt=con.createStatement();
System.out.println("Created Statement Successfully");
//Insert the record
String sql = "INSERT INTO studentdata (enroll,name) VALUES (?,
?)";
PreparedStatement istmt = con.prepareStatement(sql); istmt.setInt(1,
1);
istmt.setString(2, "aaa");
 int rowsInserted = istmt.executeUpdate();
 if (rowsInserted > 0)
{
System.out.println("A new "+rowsInserted+ " student was inserted
successfully!\n");
}
// UPDATE QUERY
PreparedStatement ustmt =con.prepareStatement("Update studentdata
set name=? where enroll =?");
```

```
ustmt.setString(1,"bbb");
ustmt.setInt(2,1);
int i=ustmt.executeUpdate();
System.out.println(i+" records updated");
System.out.println("Record Update Successfully");
//DELETE RECORD
PreparedStatement dstmt = con.prepareStatement("Delete from
studentdata where enroll =?");
dstmt.setInt(1,1);
int i1 = dstmt.executeUpdate();
System.out.println(i1+" records deleted");
System.out.println("Record Delete Successfully");
// Display the record
String sql1 = "SELECT * FROM studentdata";
 //step4 execute query
ResultSet rs=stmt.executeQuery(sql1);
while (rs.next()) {
System.out.println(rs.getInt(1)+" "+rs.getString(2));
}
//step5 close the connection object
con.close();
}
}
catch(SQLException e)
{
System.out.println(e);
```

```
}
}
}
```

OUTPUT:-

```
run:
Connection Establish Successfully
Created Statement Successfully
1 records updated
Record Update Successfully
1 bbb
2 aaa
BUILD SUCCESSFUL (total time: 0 seconds)
```

enroll	name
1	bbb
2	aaa

Practical 18

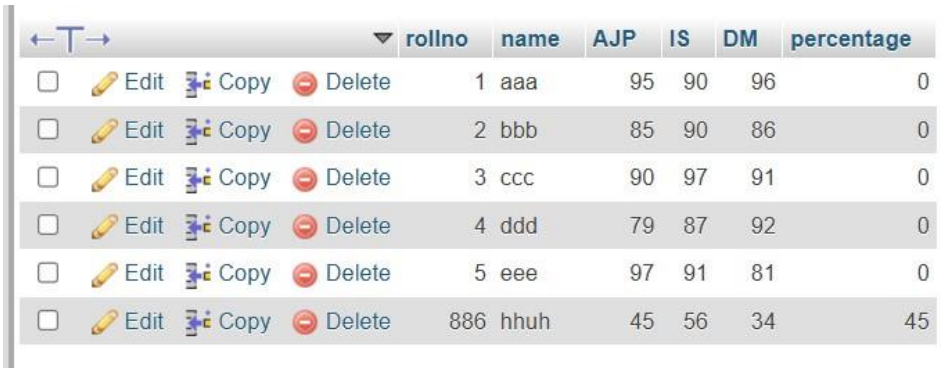
Aim: WAP to make a database student. Enter the Student detail like Enrolment no, name, and three subject marks and store all the details with percentages in the database using the JDBC driver.

```
import java.sql.*;
import java.until.scanner;
/**
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */

public class Prc18
 {
   public static void main(String[] args) throws SQLException {
       int rollno, AJP, IS, DM;
       Double Total, percentage;

 Connectionconn=
DriverManager.getConnection("jdbc:mysql://localhost:3306/ajp", "root", "");

PreparedStatement pstmt = null;
String query = "INSERT INTO prc18 (rollno, name, AJP, `IS`, DM, percentage)
VALUES (?, ?, ?, ?, ?, ?)";
```

```
try {
pstmt = conn.prepareStatement(query);
try (
Scanner sc = new Scanner(System.in))
{       System.out.println("inserting...");
System.out.println("Enter Roll No: ");
rollno = sc.nextInt();
      System.out.println("Enter name: ");
      String name = sc.next();
      System.out.println("Enter Adavance Java marks: ");
      AJP = sc.nextInt();
      System.out.println("Enter Information security marks: ");
      IS = sc.nextInt();
      System.out.println("Enter Data mining marks: ");
      DM = sc.nextInt();

      Total = (double) (AJP + IS + DM);
percentage = (Total / 300.0) * 100;
      System.out.println("percentage " + percentage);

      // Set parameters for the prepared statement
pstmt.setInt(1, rollno);
 pstmt.setString(2, name);
pstmt.setInt(3, AJP);
pstmt.setInt(4, IS);
pstmt.setInt(5, DM);
```

```
pstmt.setDouble(6, percentage);
// Bind the percentage
    }
    pstmt.executeUpdate();
    System.out.println("Data  inserted  successfully!");
}
catch (Exception e) {
  e.printStackTrace();
}         conn.close();
    }
}
```

OUTPUT:-



	rollno	name	AJP	IS	DM	percentage
Edit Copy Delete	1	aaa	95	90	96	0
Edit Copy Delete	2	bbb	85	90	86	0
Edit Copy Delete	3	ccc	90	97	91	0
Edit Copy Delete	4	ddd	79	87	92	0
Edit Copy Delete	5	eee	97	91	81	0
Edit Copy Delete	886	hhuh	45	56	34	45

```
run:
inserting...
Enter Roll No:
456
Enter name:
sdf
Enter Adavance Java marks:
48
Enter Information security marks:
34
Enter Data mining marks:
87
percentage 56.333333333333336
Data inserted successfully!
BUILD SUCCESSFUL (total time: 18 seconds)
```

Practical 19

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

```
import java.applet.Applet;
import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import java.sql.*;

/**
*@author Aman Patel
 *Name:Patel Aman Sajid bhai
*Enrollment No:2203396160199
 *Division:5QC
 * Batch:2
 */

public class Prc19 extends JFrame implements ItemListener
{
JComboBox comboProductList;
JLabel IblID, IblIDValue, IblPrice, IblPriceValue;
  JPanel p1, p2;
public String names[] = {"Choose Items..", "Desktop", "Laptop", "KeyBoard",
"Mouse", "Scanner", "Printer", "Speaker",
"Headphones"};
  public static int id = 0;
 public static int item_price = 0;
 public String Driver = "com.mysql.cj.jdbc.Driver";
public String UserName = "root";
```

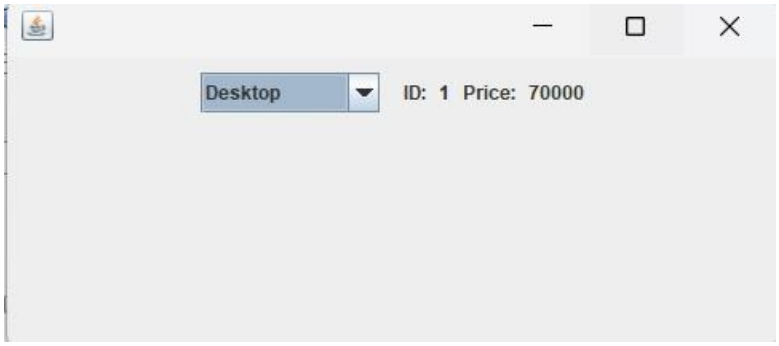
```
public String Password = "";
 public String Path = "jdbc:mysql://localhost/ajp";
  public Prc19()
{
setLayout(new FlowLayout());
p1 = new JPanel();
comboProductList = new JComboBox(names);
p1.add(comboProductList);
 p2 = new JPanel();
 IblID = new JLabel("ID: ");
IblIDValue = new JLabel("");
IblPrice = new JLabel(" Price: ");
IblPriceValue = new JLabel("");
p2.add(IblID); p2.add(IblIDValue);
p2.add(IblPrice);
p2.add(IblPriceValue);
comboProductList.addItemListener(this
);
 add(p1);  add(p2);
 setSize(500, 230);
setVisible(true);
setDefaultCloseOperation(EXIT_ON_CLOSE);
}
public void getData(String iName) throws Exception {
Class.forName(Driver);
System.out.println("Driver Establish");
Connection con = DriverManager.getConnection(Path, UserName, Password);
System.out.println("Connection Establish");
PreparedStatement ps = con.prepareStatement("Select * from items where
```

```
ItemName = ?");
ps.setString(1, iName);
ResultSet rs = ps.executeQuery();
while (rs.next())
{
id = rs.getInt("i'd");
item_price =
rs.getInt("ItemPrice");
}
System.out.println("Data Sending");
}
public void itemStateChanged(ItemEvent ie)
 {
String iName = (String)
comboProductList.getItemAt(comboProductList.getSelectedIndex()
);
 System.out.println(iName);
try
{
getData(iName);
IblIDValue.setText(String.valueOf(id));
IblPriceValue.setText(String.valueOf(item_price));
}
catch (Exception e)
{
System.out.println(e);
}
}   public static void main(String[]
args)
{
new Prc19();
}
```

}

Output:

				Id	ItemName	ItemPrice
☐	Edit	Copy	Delete	1	Laptop	70000
☐	Edit	Copy	Delete	2	Mouse	1000
☐	Edit	Copy	Delete	3	SoundBar	10000
☐	Edit	Copy	Delete	4	Scanner	50000
☐	Edit	Copy	Delete	5	Keyboard	20000
☐	Edit	Copy	Delete	6	Printer	25000
☐	Edit	Copy	Delete	7	PenDrive	8000
☐	Edit	Copy	Delete	8	Motherboard	34000




```
run:
Desktop
Driver Establish
Connection Establish
Data Sending
Desktop
Driver Establish
Connection Establish
Data Sending
```

Practical: 20

AIM: WAP to display Hello World using Servlet.

Solution:

Java Code:

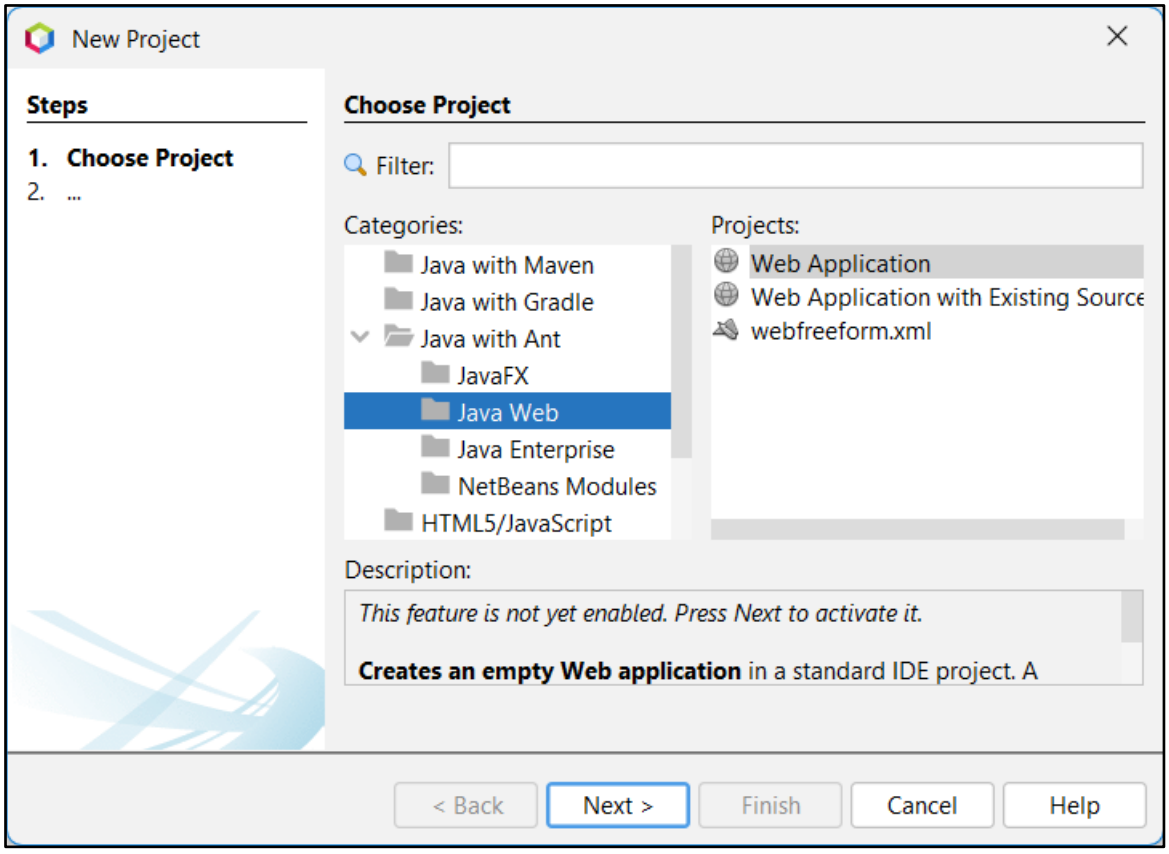
Step 1: Open Apache Netbeans IDE 15

Step 2: Click on new Project

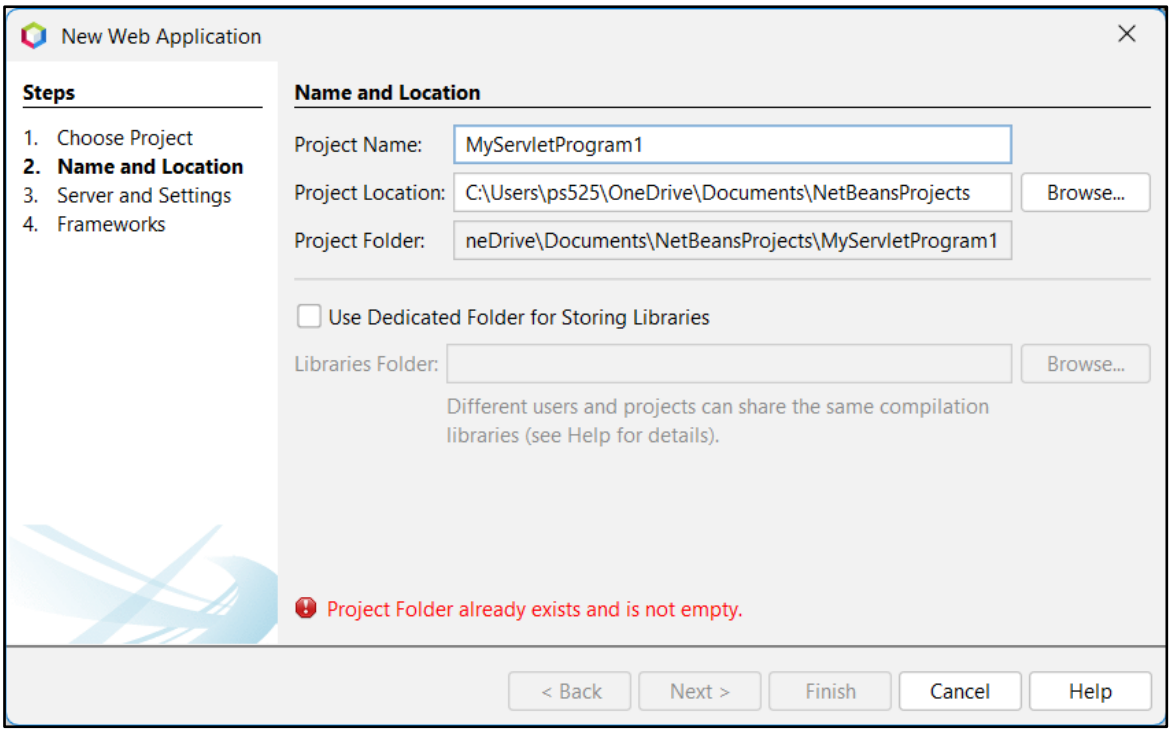
Step 3: Select Java with Ant

Step 4: Select Java web

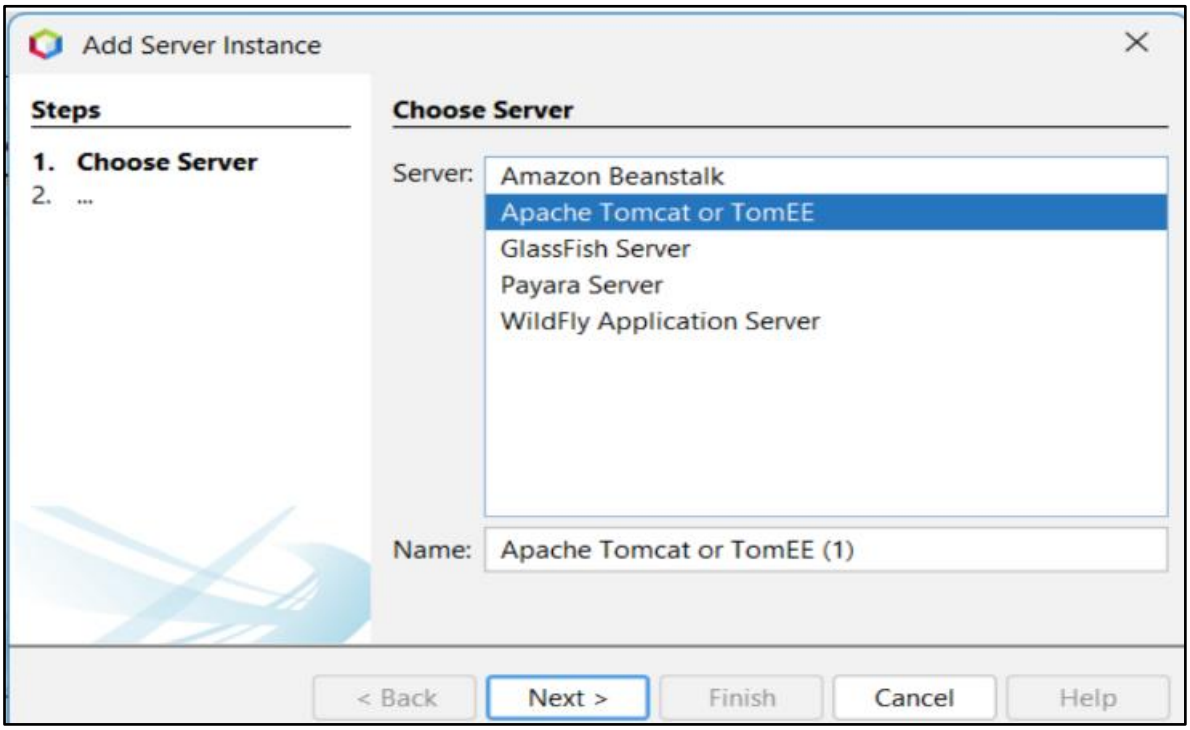
Step 5: In Java Web -> Web Application



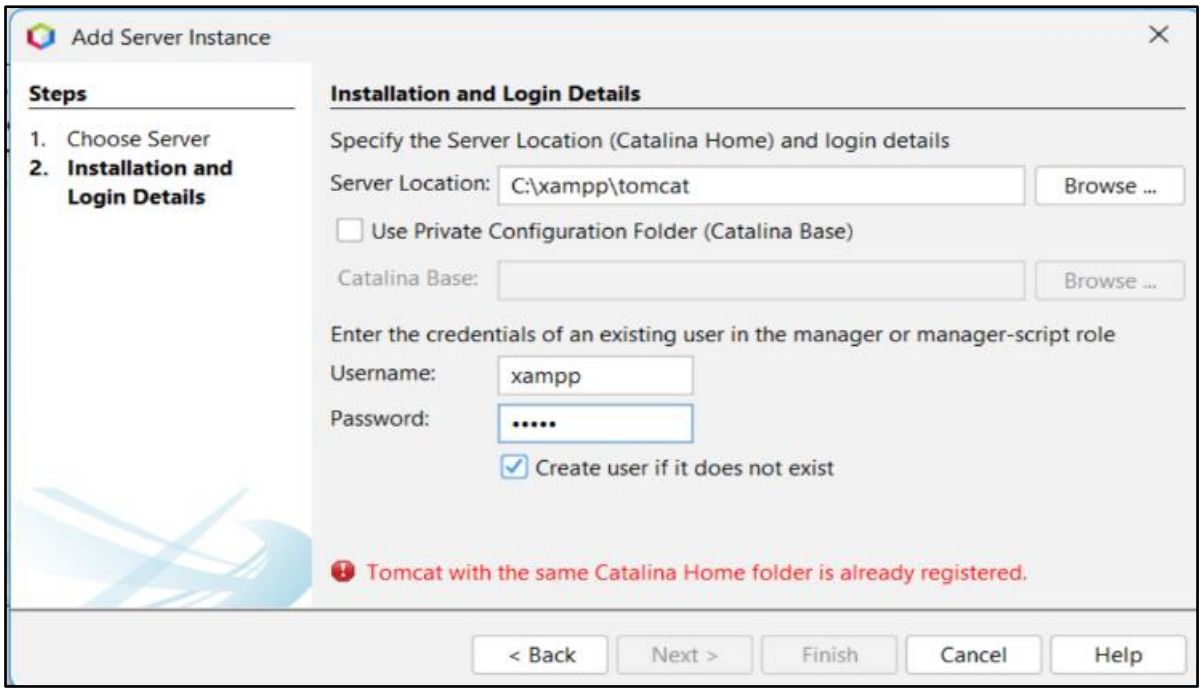
Step 6: Choose Project : After that Fill the basic information like the name of project in My case it is MyServletProgram1, second project location which by default.



Step 7: Server and Setting : In server setting, First of all choose the Server (Apache Tomcat or TomEE)



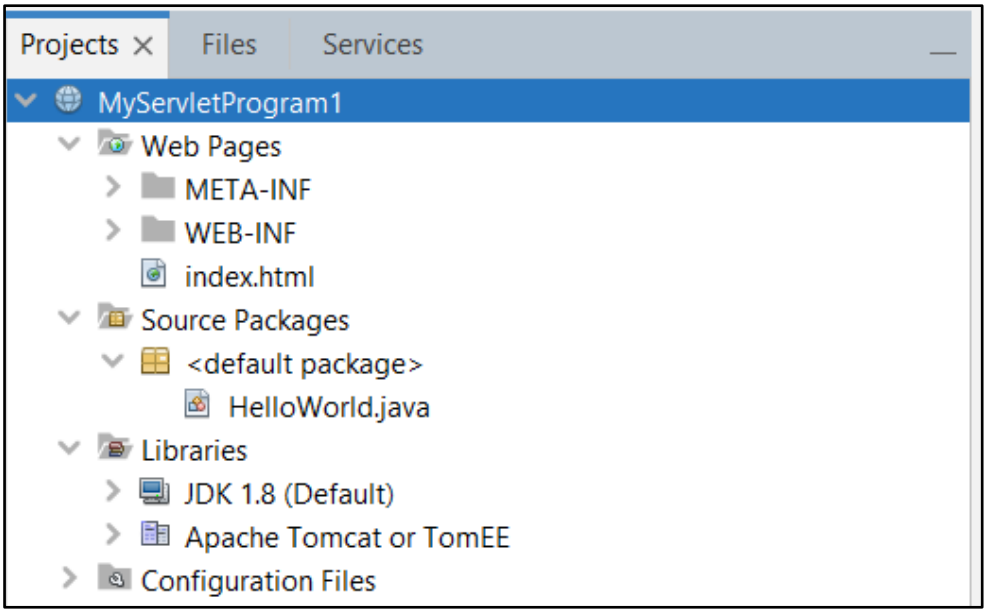
Step 8: And then choose the server location as shown below and enter password of manager or manager-script in my case it is (username - xampp and password - xampp):



Then Finish

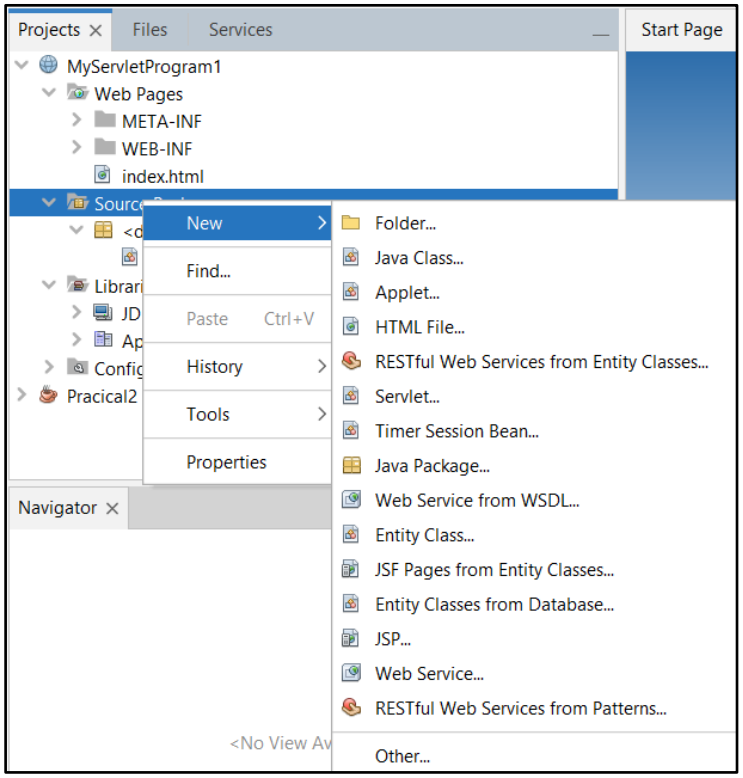
• It will create a Servlet project for you with some files that are as follow: Then Finish

• It will create a Servlet project for you with some files that are as follow:

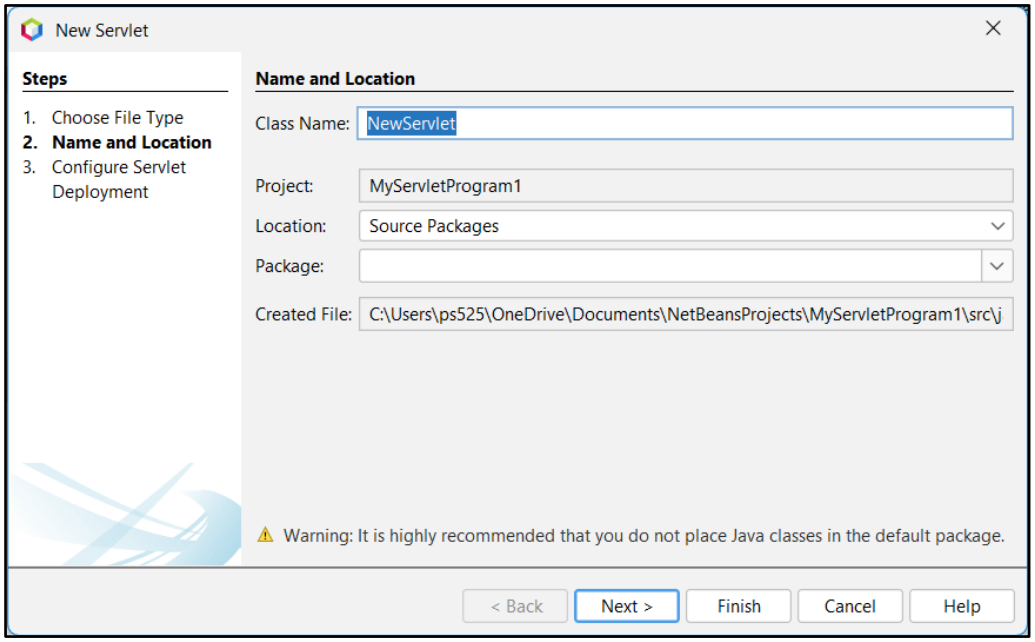


Step 8: Your project contains Web pages, Source Packages(In which we will keep our Java Code), Libraries that contain(JDK, Apache Tomcat), and Configuration Files).

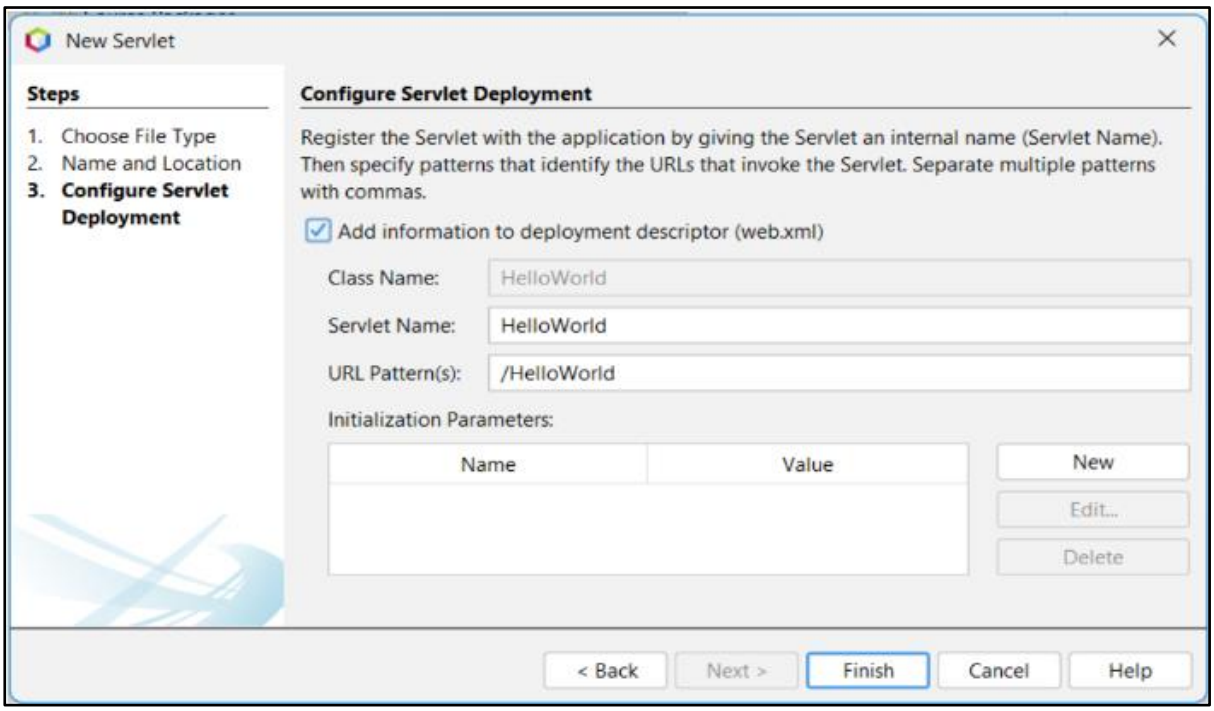
Step 9: Now Right Click on the Source Package → New →Servlet.



Step 10: Write the class name



Step 11: Configure Servlet Deployment : Tick mark on Add information to deployment descriptor (web.xml) and then click on Finish as shown below:



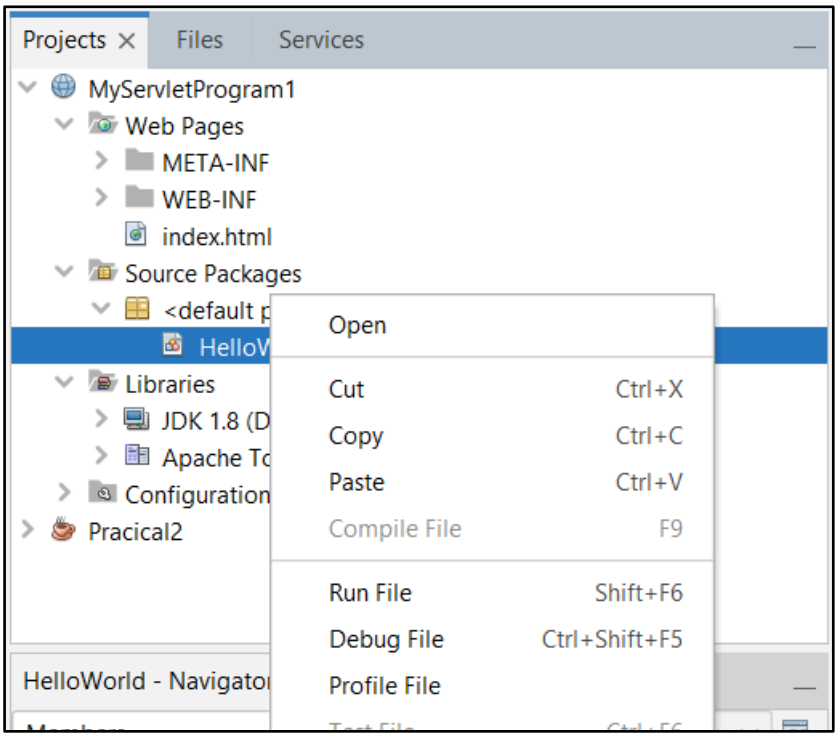
Step 12: After Finish this step it will create a HelloWorld.java file for you. Now write the below code in it

Java Code:

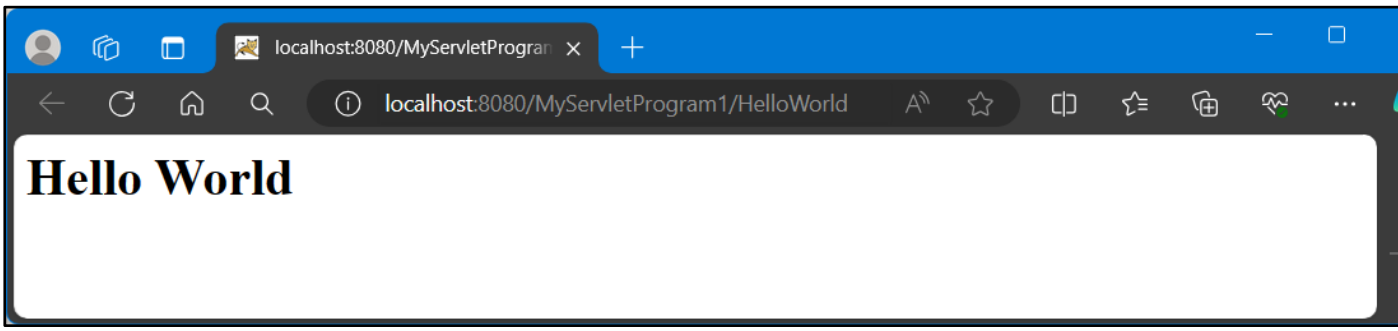
```
import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;
/**

*@author Aman Patel

 *Name:Patel Aman Sajid bhai

*Enrollment No:2203396160199
```

```
 *Division:5QC
 * Batch:2
 */
public class HelloWorld extends HttpServlet {
    private String message;
    @Override
   public void init() throws ServletException {
      // Do required initialization
      message = "Hello World";
   }
 @Override
   public void doGet(HttpServletRequest request, HttpServletResponse response)
      throws ServletException, IOException {
      // Set response content type
      response.setContentType("text/html");
// Actual logic goes here.
      PrintWriter out = response.getWriter();
      out.println("<h1>" + message + "</h1>");
   }
    @Override
   public void destroy() {
   }
}
```

Step 13: After this go to your Project Directory, Click on your Project that is MyServletProgram1 inside that go to Source Packages, then do right click on your Helloworld.java and run the file as shown below.



Step 14: After this it will ask for Set Servlet execution URL just press enter and your code will be executed on your Tomcat server.



Practical 21

Aim: Design a web page that takes the Username from user and ifit is a valid username prints "WelcomeUsername".

Java Code:-

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

 @WebServlet("/WelcomeServlet")
public class WelcomeServlet extends HttpServlet {
 private static final long serialVersionUID = 1L;
protected void doGet(HttpServletRequest request,HttpServletResponse
response) throws ServletException, IOException
 {
response.setContentType("text/html");\
PrintWriter out = response.getWriter();
out.println("<html><body>");
out.println("<h2>Welcome to the Welcome Page</h2>");
out.println("<form action='WelcomeServlet' method='post'>");
out.println("Enter Your Username: <input type='text'
name='username'required>");
out.println("<input type='submit' value='Submit'>");
```

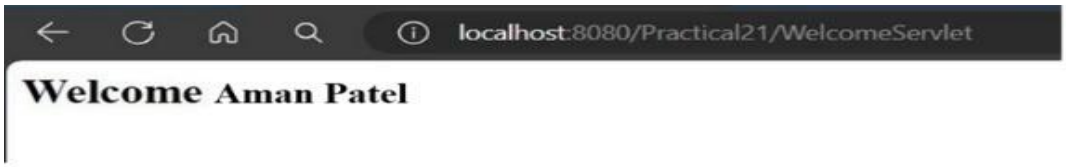
```
out.println("</form>");
out.println("</body></html>");
out.println("<h2>Invalid username format</h2>");
out.println("</body></html>");
        }
    }
boolean isValidUsername(String username) {
        // Add your validation logic here
        // For example, you could private check if the username contains only
  letters andnumbers
        // You can customize this method based on your requirements
        return username != null && username.matches("[a-zA-Z0-
        9]+");
    }
  }
```

Index.html:

```
<!DOCTYPE html>
<html>
<head>
   <title>Welcome Page</title>
</head>
<body>
   <h2>Enter Your Username</h2>
   <form action="WelcomeServlet" method="post">
      <input type="text" name="username" required>
      <input type="submit" value="Submit">
```

```
    </form>
</body>
</html>
```

OUTPUT:



Practical: 22

AIM: WAP to prepare login page using Servlet.

Solution:

Java Code:

```
import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;


@WebServlet("/LoginServlet")

public class LoginServlet extends HttpServlet {

    private static final long serialVersionUID = 1L;


    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

        response.setContentType("text/html");

        PrintWriter out = response.getWriter();


        out.println("<html><body>");

        out.println("<h2>Login</h2>");
```

```
        out.println("<form action='LoginServlet' method='post'>");
        out.println("<label for='username'>Username:</label>");
        out.println("<input type='text' id='username' name='username' required><br><br>");
        out.println("<label for='password'>Password:</label>");
        out.println("<input type='password' id='password' name='password' required><br><br>");
        out.println("<input type='submit' value='Login'>");
        out.println("</form>");
        out.println("</body></html>");
    }


 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();

        String username = request.getParameter("username");
        String password = request.getParameter("password");

        // Here, you should check the username and password against a database or some other storage
        // For simplicity, we'll just use hardcoded values
```

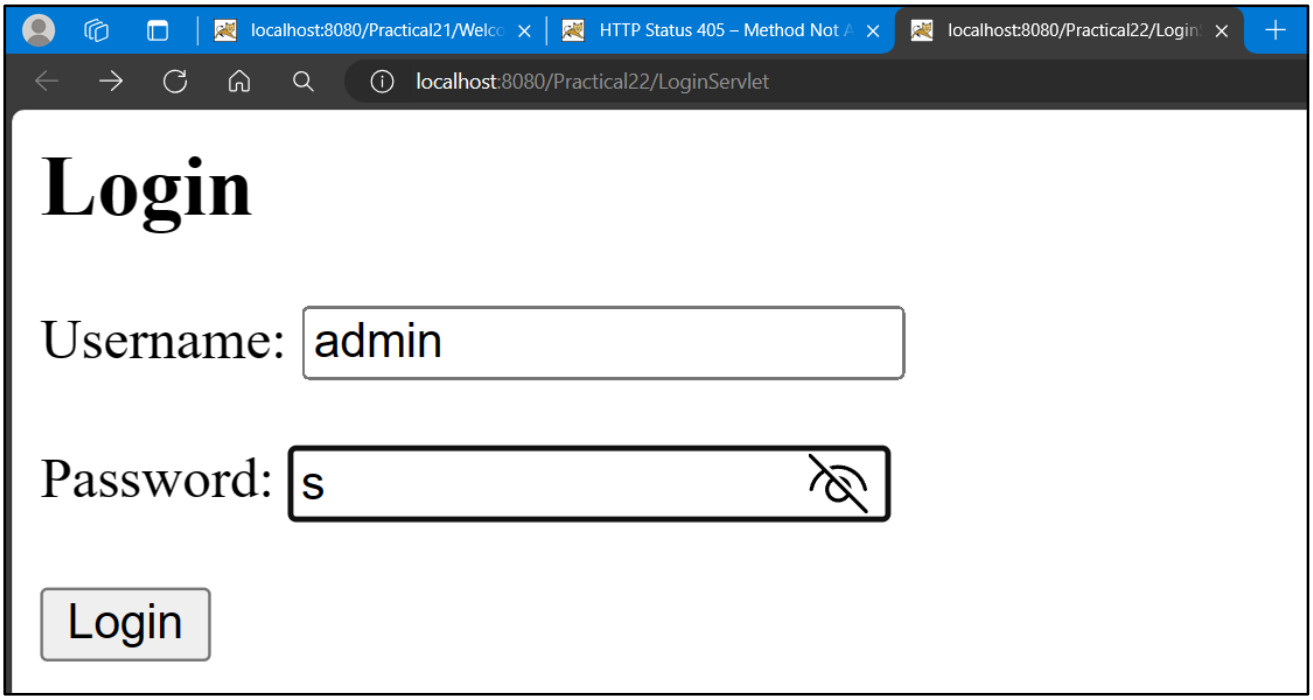
```
        if ("admin".equals(username) && "password".equals(password)) {
            out.println("<html><body>");
            out.println("<h2>Welcome, " + username + "!</h2>");
            out.println("</body></html>");
        } else {
            out.println("<html><body>");
            out.println("<h2>Invalid username or password.</h2>");
            out.println("<a href='login.html'>Try again</a>");
            out.println("</body></html>");
        }
    }
```

Index.html

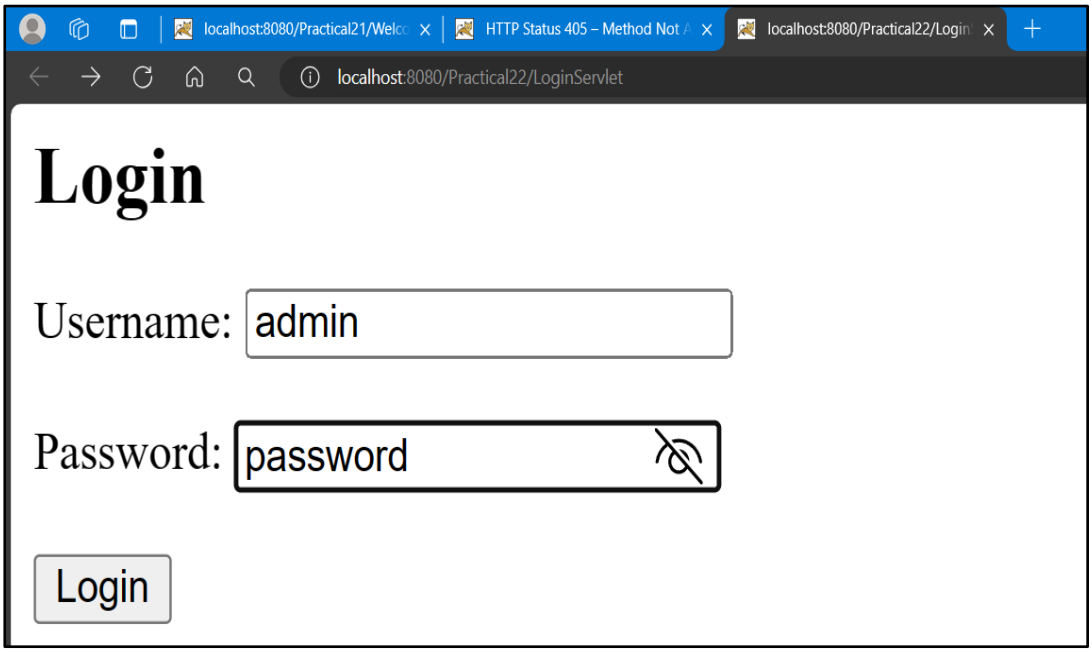
```
<!DOCTYPE html>
<html>
<head>
    <title>Login Page</title>
</head>
<body>
    <h2>Login</h2>
    <form action="LoginServlet" method="post">
```

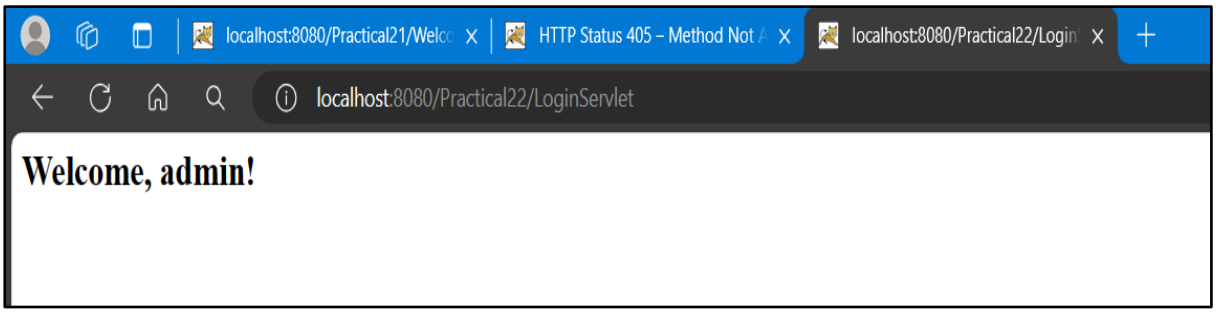
```
        <label for="username">Username:</label>
        <input type="text" id="username" name="username" required><br><br>
        <label for="password">Password:</label>
        <input type="password" id="password" name="password"
required><br><br>
        <input type="submit" value="Login">
    </form>
</body>
</html>
```

Output:




localhost:8080/Practical21/Welc
HTTP Status 405 – Method Not A
localhost:8080/Practical22/Login
localhost:8080/Practical22/LoginServlet
Invalid username or password.
Try again


localhost:8080/Practical21/Welc
HTTP Status 405 – Method Not A
localhost:8080/Practical22/Login
localhost:8080/Practical22/LoginServlet
Login
Username: admin
Password: password
Login


localhost:8080/Practical21/Welc
HTTP Status 405 – Method Not A
localhost:8080/Practical22/Login
localhost:8080/Practical22/LoginServlet
Welcome, admin!

Practical: 23

AIM: Create login form and perform state management using Cookies, HttpSession and URL Rewriting

Solution:

Java Code:

```
import java.io.IOException;

import java.io.PrintWriter;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.Cookie;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;


@WebServlet("/LoginServlet")

public class LoginServlet extends HttpServlet {

   private static final long serialVersionUID = 1L;


   protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

      response.setContentType("text/html");

      PrintWriter out = response.getWriter();
```

```
String username = request.getParameter("username");
String password = request.getParameter("password");

// Check if username and password are valid
if ("admin".equals(username) && "password".equals(password)) {
    // Create HttpSession
    HttpSession session = request.getSession();
    session.setAttribute("username", username);

    // Create Cookie for username
    Cookie usernameCookie = new Cookie("username", username);
    usernameCookie.setMaxAge(60 * 60); // Cookie lasts for 1 hour
    response.addCookie(usernameCookie);

    out.println("<html><body>");
    out.println("<h2>Welcome, " + username + "!</h2>");
    out.println("</body></html>");
} else {
    out.println("<html><body>");
    out.println("<h2>Invalid username or password.</h2>");
    out.println("<a href='login.html'>Try again</a>");
    out.println("</body></html>");
```

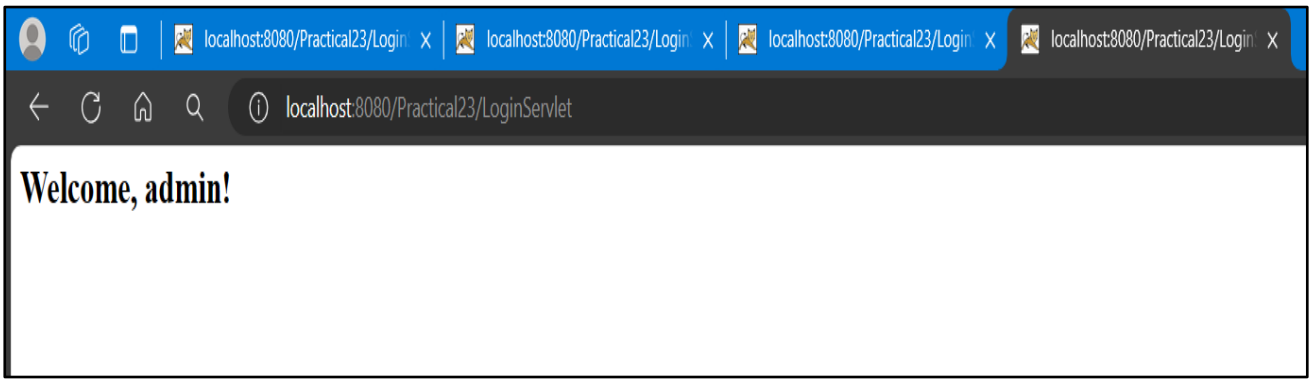
```
        }
    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        doPost(request, response);
    }
}
```

Login.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Login Page</title>
</head>
<body>
    <h2>Login</h2>
    <form action="LoginServlet" method="post">
        <label for="username">Username:</label>
        <input type="text" id="username" name="username" required><br><br>
        <label for="password">Password:</label>
        <input type="password" id="password" name="password"
required><br><br>
        <input type="submit" value="Login">
    </form>
```

```
</body>

</html>
```

Output:



Practical: 24

AIM: WAP to display Hello World using JSP.

Solution:

Java Code:

```
<html>
    <head>
        <title>Welcome</title>
    </head>
    <body>
        <h1>Hello World.</h1>
    </body>
</html>
```

Output: