PRACTICAL-1

Aim:- Write a program to create registration form for the student using AWT.

Code:

```
import java.awt.*;
class MyForm extends Frame
{ MyForm()
  {   Frame fm=new Frame();
      setSize(800,800);
      setVisible(true);
      setLayout(null);
      Label n=new Label("Hemil Chovatiya");
      n.setBounds(200,45,200,30);
      Label n1=new Label("Name");
      n1.setBounds(20,105,110,20);
      TextField na=new TextField(20);
      na.setBounds(140,105,200,20);
       Label n2=new Label("Enrollment No.");
      n2.setBounds(20,140,110,20);
      TextField naa=new TextField(20);
      naa.setBounds(140,140,200,20);
      Label g=new Label("Sem");
      g.setBounds(20,175,110,20);
      Choice c=new Choice();
      c.setBounds(140,175,200,20);
      c.add("3");
      c.add("5");
      c.add("7");
      Label n3=new Label("Gender");
      n3.setBounds(20,210,110,20);
      CheckboxGroup cbg =new CheckboxGroup();
      Checkbox r1=new Checkbox("Male",cbg,false);
      r1.setBounds(140,210,50,20);
      Checkbox r2=new Checkbox("female",cbg,false);
      r2.setBounds(200,210,70,20);
      Label n4=new Label("Subject");
      n4.setBounds(20,245,110,20);
      Checkbox c1=new Checkbox("php");
      c1.setBounds(140,245,50,20);
      Checkbox c2=new Checkbox("Android");
      c2.setBounds(200,245,70,20);
```

```
        Checkbox c3=new Checkbox("C++");
        c3.setBounds(280,245,50,20);
        Checkbox c4=new Checkbox("Swift");
        c4.setBounds(340,245,50,20);
        Label n5=new Label("Suggestion ");
        n5.setBounds(20,280,110,20);
        TextArea na1=new TextArea();
        na1.setBounds(140,277,300,80);
        Button b1=new Button("Submit");
        b1.setBounds(140,380,110,25);
        Button b2=new Button("cancel");
        b2.setBounds(260,380,110,25);
        add(b1);
        add(b2);
        add(n5);
        add(na1);
        add(r1);
        add(r2);
        add(g);
        add(n);
        add(naa);
        add(na);
        add(n1);
        add(n2);
        add(n3);
        add(n4);
        add(c);
        add(c1);
        add(c2);
        add(c3);
        add(c4);          }
 public static void main(String args[])
      {      MyForm mf=new MyForm();        }
}
```
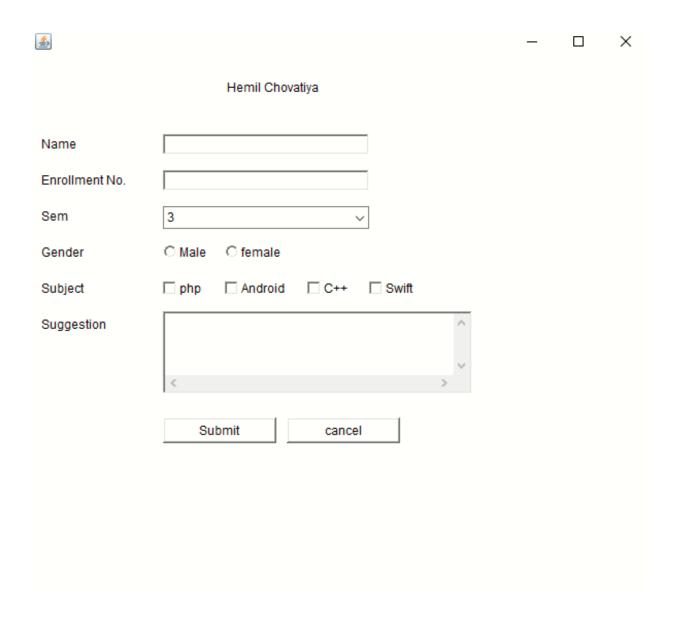
PRACTICAL-2

Aim:- Write a program to create registration form for the student using swing.

Code:

```java
import java.awt.*;
import javax.swing.*;
public class Myform extends JFrame
{
   public  Myform()
  {
     setSize(800,800);
     setVisible(true);
     setLayout(null);
     JLabel n=new JLabel("Hemil Chovatiya Registration Form");
     n.setBounds(200,45,400,20);
     n.setFont(new Font("arial",Font.PLAIN,20));
     JLabel n1=new JLabel("Name");
     n1.setFont(new Font("arial",Font.PLAIN,15));
     n1.setBounds(20,105,110,20);
     JTextField na=new JTextField(20);
     na.setBounds(140,105,200,20);
     JLabel n2=new JLabel("Enrollment No.");
     n2.setFont(new Font("arial",Font.PLAIN,15));
     n2.setBounds(20,140,110,20);
     JTextField naa=new JTextField(20);
     naa.setBounds(140,140,200,20);
     JLabel g=new JLabel("Sem");
     g.setFont(new Font("arial",Font.PLAIN,15));
     g.setBounds(20,175,110,20);
     JComboBox c=new JComboBox<>();
 c.setModel(new DefaultComboBoxModel<>(new String[] { "Sem-3", "Sem-5", "Sem-7" }));
     c.setFont(new Font("arial",Font.PLAIN,15));
     c.setBounds(140,175,200,20);
     JLabel n3=new JLabel("Gender");
     n3.setFont(new Font("arial",Font.PLAIN,15));
     n3.setBounds(20,210,110,20);
     JRadioButton r1=new JRadioButton("Male",false);
     r1.setFont(new Font("arial",Font.PLAIN,15));
     ButtonGroup cbg=new ButtonGroup();
     r1.setBounds(140,210,70,20);
```

```java
JRadioButton r2=new JRadioButton("Female",false);
r2.setFont(new Font("arial",Font.PLAIN,15));
r2.setBounds(210,210,70,20);
cbg.add(r1);
cbg.add(r2);
JLabel n4=new JLabel("Subject");
n4.setFont(new Font("arial",Font.PLAIN,15));
n4.setBounds(20,245,110,20);
JRadioButton x=new JRadioButton();
JCheckBox c1=new JCheckBox("php");
c1.setFont(new Font("arial",Font.PLAIN,15));
c1.setBounds(140,245,60,20);
 JCheckBox c2=new JCheckBox("Android");
c2.setFont(new Font("arial",Font.PLAIN,15));
c2.setBounds(200,245,83,20);
JCheckBox c3=new JCheckBox("C++");
c3.setFont(new Font("arial",Font.PLAIN,15));
c3.setBounds(280,245,60,20);
JCheckBox c4=new JCheckBox("Swift");
c4.setFont(new Font("arial",Font.PLAIN,15));
c4.setBounds(340,245,70,20);
JLabel n6=new JLabel("Password ");
n6.setFont(new Font("arial",Font.PLAIN,15));
n6.setBounds(20,280,110,20);
JPasswordField pf=new JPasswordField("Enter Password");
pf.setFont(new Font("arial",Font.PLAIN,15));
pf.setBounds(140,280,200,20);
JLabel n5=new JLabel("Suggestion ");
n5.setFont(new Font("arial",Font.PLAIN,15));
n5.setBounds(20,315,110,20);
JTextArea na1=new JTextArea();
na1.setBounds(140,315,300,80);
JButton b1=new JButton("Submit");
b1.setFont(new Font("arial",Font.PLAIN,15));
b1.setBounds(140,410,110,25);
JButton b2=new JButton("Cancel");
b2.setFont(new Font("arial",Font.PLAIN,15));
b2.setBounds(260,410,110,25);
add(pf);
add(n6);
add(b1);
add(b2);
add(n5);
```

```
        add(na1);
        add(r1);
        add(r2);
        add(g);
        add(n);
        add(naa);
        add(na);
        add(n1);
        add(n2);
        add(n3);
        add(n4);
        add(c);
        add(c1);
        add(c2);
        add(c3);
        add(c4);        }
    public static void main(String args[])
    {       Myform mf=new Myform();       }
}
```
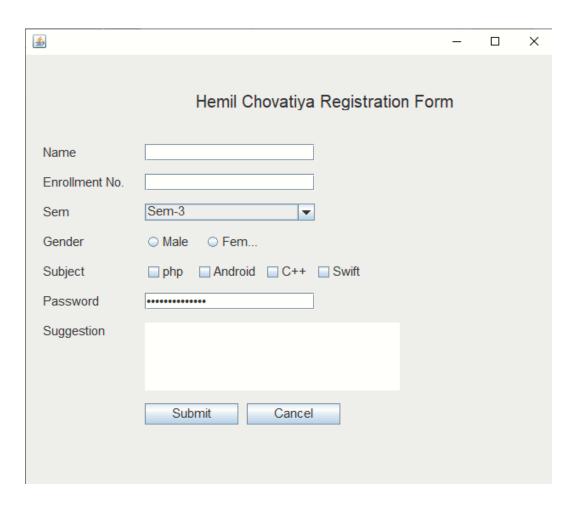
Output:

Hemil Chovatiya Registration Form

Name	
Enrollment No.	
Sem	Sem-3
Gender	○ Male ○ Fem...
Subject	☐ php ☐ Android ☐ C++ ☐ Swift
Password	••••••••••••
Suggestion	

Submit   Cancel

PRACTICAL-3

Aim:- Implement JDBC by connecting with database and execute Prepared Statement

Code:

```
import java.util.*;
import java.sql.*;
class crud {
    Scanner sc = new Scanner(System.in);
    void delete() {
        System.out.println("Delete Row");
        System.out.println("Enter uname to delete row: ");
        String uname = sc.nextLine();
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:mysql://localhost:3306/farm", "root", "mysql");
            String qry = "delete from login where uname=?";
            PreparedStatement ps = con.prepareStatement(qry);
            ps.setString(1, uname);
            int rs = ps.executeUpdate();
            if (rs >= 1) {
                System.out.println(rs+" Row deleted successfuly");
            } else {
                System.out.println(rs+" Row deleted successfuly");
            }
            ps.close();
            con.close();
        } catch (Exception e) {
            System.out.println(e);
        }
    }
    void update() {
        System.out.println("Update Rows");
        System.out.println("Enter uname which you want to update row: ");
        String uname = sc.nextLine();
        System.out.println("Enter fname: ");
        String fname = sc.nextLine();
        System.out.println("Enter lname: ");
        String lname = sc.nextLine();
        System.out.println("Enter dob(YYYY-MM-DD): ");
```

```java
        String dob = sc.nextLine();
        System.out.println("Enter password: ");
        String password = sc.nextLine();
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con = DriverManager.getConnection(
                    "jdbc:mysql://localhost:3306/farm", "root", "mysql");
            String qry = "update login set fname=?,lname=?,dob=?,pwd=? where uname=?";
            PreparedStatement ps = con.prepareStatement(qry);
            ps.setString(1, fname);
            ps.setString(2, lname);
            ps.setString(3, dob);
            ps.setString(4, password);
            ps.setString(5, uname);
            int rs = ps.executeUpdate();
            if (rs >= 1) {
                System.out.println(rs+"Rows Updated successfuly");
            } else {
                System.out.println(rs+"Rows Updated successfuly");
            }
            ps.close();
            con.close();
        } catch (Exception e) {
            System.out.println(e);
        }
    }
    void insert() {
        System.out.println("Enter ");
        System.out.println("Enter fname: ");
        String fname = sc.nextLine();
        System.out.println("Enter lname: ");
        String lname = sc.nextLine();
        System.out.println("Enter uname: ");
        String uname = sc.nextLine();
        System.out.println("Enter dob(YYYY-MM-DD): ");
        String dob = sc.nextLine();
        System.out.println("Enter password: ");
        String password = sc.nextLine();
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con = DriverManager.getConnection(
                    "jdbc:mysql://localhost:3306/farm", "root", "mysql");
```

```java
        String qry = "insert into login values(?,?,?,?,?)";
        PreparedStatement ps = con.prepareStatement(qry);
        ps.setString(1, fname);
        ps.setString(2, lname);
        ps.setString(3, uname);
        ps.setString(4, dob);
        ps.setString(5, password);
        int rs = ps.executeUpdate();
        if (rs >= 1) {
            System.out.println(rs+" Rows Inserted successfuly");
        } else {
            System.out.println(rs+" Rows Inserted successfuly");
        }
        ps.close();
        con.close();
    } catch (Exception e) {
        System.out.println(e);
    }
    }
    void display() {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/farm",
"root", "mysql");
            String qry = "select * from login;";
            Statement stmt = con.createStatement();
            ResultSet rs = stmt.executeQuery(qry);
            System.out.println("fname \t lname \t uname \t dob \t password");
            while (rs.next()) {
                System.out.println("" + rs.getString(1) + "\t" + rs.getString(2) +
                "\t" + rs.getString(3) + "\t"+ rs.getString(4) + "\t" + rs.getString(5));
            }
            stmt.close();
            con.close();
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
public class App {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("CRUD operation by hemil on mysql Database");
```

```java
        crud isrt = new crud();
        int ch = 0;
        while (ch <= 4) {
            System.out.println("1.Insert Data");
            System.out.println("2.Update Data");
            System.out.println("3.Delete Data");
            System.out.println("4.Display Data");
            System.out.println("5.Exit");
            System.out.println("Enter Choice :");
            ch = sc.nextInt();
            switch (ch) {
                case 1:
                    isrt.insert();
                    break;
                case 2:
                    isrt.update();
                    break;
                case 3:
                    isrt.delete();
                    break;
                case 4:
                    isrt.display();
                    break;
                default:
                    System.out.println("Defafult choice error");
                    break;
            }
        }
        sc.close();
    }
}
```

Output:

```
PS F:\pu_it_pratical\Adv. Java\p3_ps_insrt_up
\bin\java.exe' '-XX:+ShowCodeDetailsInExcepti
CRUD operation by hemil on mysql Database
1.Insert Data
2.Update Data
3.Delete Data
4.Display Data
5.Exit
Enter Choice :
1
Insert Values:
Enter fname:
Hemil Chovatiya
Enter lname:
Chovatiya
Enter uname:
HemilCHovatiya4@gmail.com
Enter dob(YYYY-MM-DD):
2001-07-20
Enter password:
HemilChovatiya
1 Rows Inserted successfuly
1.Insert Data
2.Update Data
3.Delete Data
4.Display Data
5.Exit
Enter Choice :
2
Updates values in Rows
Enter uname which you want to update row:
HemilCHovatiya4@gmail.com
Enter fname:
Hemil
Enter lname:
Chovatiya
Enter dob(YYYY-MM-DD):
```

```
HemilCHovatiya4@gmail.com
Enter dob(YYYY-MM-DD):
2001-07-20
Enter password:
HemilChovatiya
1 Rows Inserted successfuly
1.Insert Data
2.Update Data
3.Delete Data
4.Display Data
5.Exit
Enter Choice :
2
Updates values in Rows
Enter uname which you want to update row:
HemilCHovatiya4@gmail.com
Enter fname:
Hemil
Enter lname:
Chovatiya
Enter dob(YYYY-MM-DD):
2001-07-20
Enter password:
Hemil
1Rows Updated successfuly
1.Insert Data
2.Update Data
3.Delete Data
4.Display Data
5.Exit
Enter Choice :
3
Delete Row
Enter uname to delete row:
HemilCHovatiya4@gmail.com
1 Row deleted successfuly
1.Insert Data
```

```
Enter Choice :
3
Delete Row
Enter uname to delete row:
HemilCHovatiya4@gmail.com
1 Row deleted successfuly
1.Insert Data
2.Update Data
3.Delete Data
4.Display Data
5.Exit
Enter Choice :
4
fname    lname    uname    dob         password
2121     1221     212      2001-01-01      123
212      21212    2212121  2001-01-21      12
Bhaumik Patel    Bhaumik@200     2002-04-08      Bhaumik@022
dharmik chovatiya        dc@gmail.com    2001-11-01      vdvd
fenil    gondliya        fenil@gmail.com 2001-06-11      fg
hemil    chovatiya       hemil@20gmail   2001-05-04      hemil
JAY      KHANPARA        JAYKHANPARA4@GMAIL.COM  2003-01-04      Jk@2000
kemil    thummar kenil@gmail.com 2001-07-07      kEMIL
Krish    gajera  kg@gmail.com    2002-10-21      kg
krit     Kakadiya        kk@gmail.com    2005-04-04      kkad@20
romit    ghevariya       RG@gmail.com    2002-01-01      Rg@2332
TIRTH    KANERIYA        TIRTH@2100      2002-07-07      tirth@02
Utsav    Bhanderi        utb@2903        2002-08-04      utu@240
1.Insert Data
2.Update Data
3.Delete Data
4.Display Data
5.Exit
Enter Choice :
5
Defafult choice error
PS F:\pu it pratical\Adv. Java\p3 ps insrt update delete>
```

fname	lname	uname	dob	pwd
2121	1221	212	2001-01-01	123
212	21212	2212121	2001-01-21	12
Bhaumik	Patel	Bhaumik@200	2002-04-08	Bhaumik@022
dharmik	chovatiya	dc@gmail.com	2001-11-01	vdvd
fenil	gondliya	fenil@gmail.com	2001-06-11	fg
hemil	chovatiya	hemil@20gmail	2001-05-04	hemil
JAY	KHANPARA	JAYKHANPARA4@GMAIL.COM	2003-01-04	Jk@2000
kemil	thummar	kenil@gmail.com	2001-07-07	kEMIL
Krish	gajera	kg@gmail.com	2002-10-21	kg
krit	Kakadiya	kk@gmail.com	2005-04-04	kkad@20
romit	ghevariya	RG@gmail.com	2002-01-01	Rg@2332
TIRTH	KANERIYA	TIRTH@2100	2002-07-07	tirth@02
Utsav	Bhanderi	utb@2903	2002-08-04	utu@240
NULL	NULL	NULL	NULL	NULL

PRACTICAL-4

Aim: Implement JDBC by connecting with database and execute Callable Statement.

Code:

```java
import java.sql.*;
import java.util.*;
public class App{
public static void main(String[] args)
{    Scanner sc=new Scanner(System.in);
     System.out.println("Insert Values:");
     System.out.println("Enter fname: ");
     String fname = sc.nextLine();
     System.out.println("Enter lname: ");
     String lname = sc.nextLine();
     System.out.println("Enter uname: ");
     String uname = sc.nextLine();
     System.out.println("Enter dob(YYYY-MM-DD): ");
     String dob = sc.nextLine();
     System.out.println("Enter password: ");
     String password = sc.nextLine();
     try {
        Class.forName("com.mysql.cj.jdbc.Driver");
        Connection con = DriverManager.getConnection(
            "jdbc:mysql://localhost:3306/farm", "root", "mysql");

        String qry = "insert into login values(?,?,?,?,?)";
        CallableStatement cst = con.prepareCall(qry);
        cst.setString(1, fname);
        cst.setString(2, lname);
     cst.setString(3, uname);
     cst.setString(4, dob);
        cst.setString(5, password);
        int rs = cst.executeUpdate();
        if (rs >= 1) {
           System.out.println(rs+" Rows Inserted successfuly");
        } else {
           System.out.println(rs+" Rows Inserted successfuly");
        }
        cst.close();
        con.close();
```
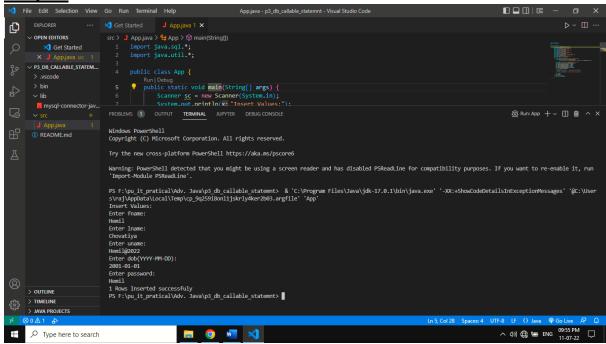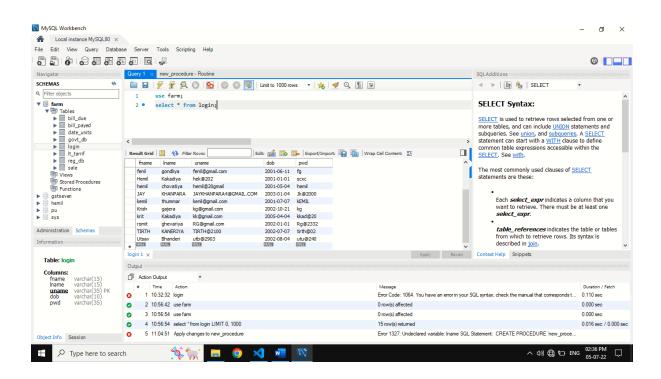
```
} catch (Exception e) {
    System.out.println(e);
}            }            }
```

Output:

PRACTICAL-5

Aim: Implement chat application using java.net

Code:chat_client.java

```java
package chat;
import static chat.chat_server.dout;
import static chat.chat_server.ss;
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.net.ServerSocket;
import java.net.Socket;

public class chat_client extends javax.swing.JFrame {
    static Socket s;
    static DataInputStream dis;
    static DataOutputStream dout;
    public chat_client() {
        initComponents();
    }
    @SuppressWarnings("unchecked")
BEGIN:initComponents
    private void initComponents() {

        jScrollPane1 = new javax.swing.JScrollPane();
        msg_area = new javax.swing.JTextArea();
        msg_text = new javax.swing.JTextField();
        msg_send = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        msg_area.setColumns(20);
        msg_area.setRows(5);
        jScrollPane1.setViewportView(msg_area);
        msg_send.setText("send");
        msg_send.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                msg_sendActionPerformed(evt);
            }
        });
        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
        jLabel1.setText("client");
        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
```

```java
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jScrollPane1)
        .addGroup(layout.createSequentialGroup()
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addComponent(msg_text,
javax.swing.GroupLayout.PREFERRED_SIZE, 316,
javax.swing.GroupLayout.PREFERRED_SIZE)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(msg_send))
                .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE,
127, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(0, 18, Short.MAX_VALUE)))
        .addContainerGap())    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
        .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 23,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 224,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(msg_text)
            .addGroup(layout.createSequentialGroup()
                .addComponent(msg_send)
                .addGap(0, 2, Short.MAX_VALUE)))
        .addContainerGap())
    );
    pack();
    }// </editor-fold>//GEN-END:initComponents
    private void msg_sendActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_msg_sendActionPerformed
     try{
    String msg="";
    msg=msg_text.getText();
    dout.writeUTF(msg);
    msg_text.setText("");        }
```

```java
        catch(Exception e)
        {       //handle the exception here      }
    }//GEN-LAST:event_msg_sendActionPerformed
    public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
     * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
     */
    try {
        for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
            if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
            }
        }
    } catch (ClassNotFoundException ex) {
java.util.logging.Logger.getLogger(chat_client.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (InstantiationException ex) {
java.util.logging.Logger.getLogger(chat_client.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (IllegalAccessException ex) {
java.util.logging.Logger.getLogger(chat_client.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    } catch (javax.swing.UnsupportedLookAndFeelException ex) {
java.util.logging.Logger.getLogger(chat_client.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
    //</editor-fold>
    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new chat_client().setVisible(true);
        }
    });
    try {
        String msgin = "";
        s = new Socket("127.0.0.1",1201); // ip address is of localhost because server is
running on the same machine
```

```
        dis = new DataInputStream(s.getInputStream());
        dout = new DataOutputStream(s.getOutputStream());
        while (!msgin.equals("exit")) {
            msgin = dis.readUTF();
            msg_area.setText(msg_area.getText() + "\n Server : " + msgin);
        }
    } catch (Exception e) {          //handle the exception here        }
}   // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JLabel jLabel1;
    private javax.swing.JScrollPane jScrollPane1;
    private static javax.swing.JTextArea msg_area;
    private javax.swing.JButton msg_send;
    private javax.swing.JTextField msg_text;
    // End of variables declaration//GEN-END:variables     }
```

chat_server.java

```
package chat;
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.net.ServerSocket;
import java.net.Socket;

public class chat_server extends javax.swing.JFrame {
    static ServerSocket ss;
    static Socket s;
    static DataInputStream dis;
    static DataOutputStream dout;
    /**    * Creates new form chat_server    */
    public chat_server() {
        initComponents();
    }
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-
BEGIN:initComponents
    private void initComponents() {

        jScrollPane1 = new javax.swing.JScrollPane();
        msg_area = new javax.swing.JTextArea();
        msg_text = new javax.swing.JTextField();
        msg_send = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        msg_area.setColumns(20);
        msg_area.setRows(5);
```

```java
      jScrollPane1.setViewportView(msg_area);
      msg_send.setText("send");
      msg_send.addActionListener(new java.awt.event.ActionListener() {
         public void actionPerformed(java.awt.event.ActionEvent evt) {
            msg_sendActionPerformed(evt);
         }
      });
      jLabel1.setFont(new java.awt.Font("Tahoma", 1, 18)); // NOI18N
      jLabel1.setText("Server");
      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(jScrollPane1)
            .addGroup(layout.createSequentialGroup()
.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                  .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE,
114, javax.swing.GroupLayout.PREFERRED_SIZE)
               .addGroup(layout.createSequentialGroup()
                  .addComponent(msg_text,
javax.swing.GroupLayout.PREFERRED_SIZE, 303,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                  .addComponent(msg_send)))
               .addGap(0, 18, Short.MAX_VALUE)))
            .addContainerGap())
      );
      layout.setVerticalGroup(
         layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
         .addGroup(layout.createSequentialGroup()
            .addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED_SIZE, 24,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED_SIZE, 225,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
               .addComponent(msg_text)
```

```java
            .addComponent(msg_send, javax.swing.GroupLayout.DEFAULT_SIZE, 28,
Short.MAX_VALUE))
            .addContainerGap())
    );
    pack();
  }// </editor-fold>//GEN-END:initComponents
  private void msg_sendActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_msg_sendActionPerformed
      try{
      String msg="";
      msg=msg_text.getText();
      dout.writeUTF(msg);
      msg_text.setText("");
      }
      catch(Exception e)
      {       //handle the exception here      }
}//GEN-LAST:event_msg_sendActionPerformed
  public static void main(String args[]) {
      /* Set the Nimbus look and feel */
      //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
/* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
*For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html   */
      try {
         for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
             if ("Nimbus".equals(info.getName())) {
                javax.swing.UIManager.setLookAndFeel(info.getClassName());
                break;
             }
         }
      } catch (ClassNotFoundException ex) {
java.util.logging.Logger.getLogger(chat_server.class.getName()).log(java.util.logging.Level.
SEVERE, null, ex);
      } catch (InstantiationException ex) {
java.util.logging.Logger.getLogger(chat_server.class.getName()).log(java.util.logging.Level.
SEVERE, null, ex);
      } catch (IllegalAccessException ex) {
java.util.logging.Logger.getLogger(chat_server.class.getName()).log(java.util.logging.Level.
SEVERE, null, ex);
      } catch (javax.swing.UnsupportedLookAndFeelException ex) {
java.util.logging.Logger.getLogger(chat_server.class.getName()).log(java.util.logging.Level.
SEVERE, null, ex);
      }       //</editor-fold>
```
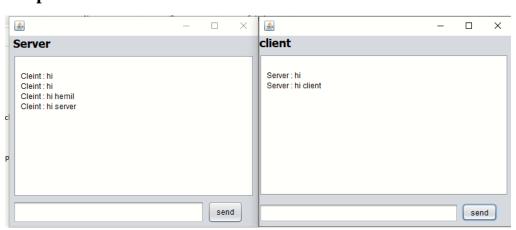
```java
/* Create and display the form */
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new chat_server().setVisible(true);
    }
});
try {
    String msgin = "";
    ss = new ServerSocket(1201);
    s = ss.accept();
    dis = new DataInputStream(s.getInputStream());
    dout = new DataOutputStream(s.getOutputStream());

    while (!msgin.equals("exit")) {
        msgin = dis.readUTF();
        msg_area.setText(msg_area.getText() + "\n Client : " + msgin);
    }
} catch (Exception e) {          //handle the exception here        }
}
// Variables declaration - do not modify//GEN-BEGIN:variables
private javax.swing.JLabel jLabel1;
private javax.swing.JScrollPane jScrollPane1;
private static javax.swing.JTextArea msg_area;
private javax.swing.JButton msg_send;
private javax.swing.JTextField msg_text;
// End of variables declaration//GEN-END:variables     }
```

Output:

PRACTICAL-6

Aim: Implement any one sorting algorithm using TCP/UDP on Server application and Give Input on Client side and client should sorted output from server and display sorted on input side.

Code:

File: SortClient.java
```java
package sort;
import java.io.*;
import java.net.*;
class SortClient
{
    public static void main(String ar[]) throws Exception
    {
        Socket s=new Socket("localhost",12345);
        PrintWriter p=new PrintWriter(s.getOutputStream());
        BufferedReader in=new BufferedReader(new InputStreamReader(s.getInputStream()));
        BufferedReader ink=new BufferedReader(new InputStreamReader(System.in));
        System.out.println("How many numbers to sort? ");
        int num=Integer.parseInt(ink.readLine());
        p.println(num);
        p.flush();
        System.out.println("Enter "+num+" numbers to sort :");
        String sarr[]=new String[num];
        for(int i=0;i<num;i++)
        {
            System.out.print("no. "+i+"=");
            sarr[i]=ink.readLine();
            p.println(sarr[i]);
            p.flush();
        }
        String res;
        System.out.println("\nSorted array::\n");
        while((res=in.readLine())!=null)
        {
            System.out.println(res);
        }
        s.close();
    }
}
```

File: SortServer.java

```java
package sort;
import java.io.*;
import java.net.*;
class SortServer
{
    public static void main(String ar[]) throws Exception
    {
        ServerSocket s1=new ServerSocket(12345);
        System.out.println(" Server Started");
        Socket s=s1.accept();
            PrintWriter p=new PrintWriter(s.getOutputStream());
            BufferedReader in=new BufferedReader(new
          InputStreamReader(s.getInputStream()));
            String num=in.readLine();
            int n=Integer.parseInt(num);
            System.out.println("Client want to sort "+n+" numbers");
        String sarr[]=new String[n];
        int arr[]=new int[n];
        int swap,c,d;
        System.out.println("received numbers::\n");
        for(int i=0;i<n;i++)
        {
            sarr[i]=in.readLine();
            arr[i]=Integer.parseInt(sarr[i]);
            System.out.println("no. "+i+"="+arr[i]);

        }

        for (c = 0; c < ( n - 1 ); c++)
        {
            for (d = 0; d < n - c - 1; d++)
            {
                if (arr[d] > arr[d+1])
                {
                    swap    = arr[d];
                    arr[d]   = arr[d+1];
                    arr[d+1] = swap;
                }
            }
        }

        System.out.println("\nSorted list of numbers");
```
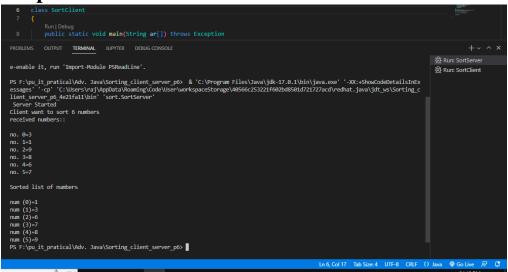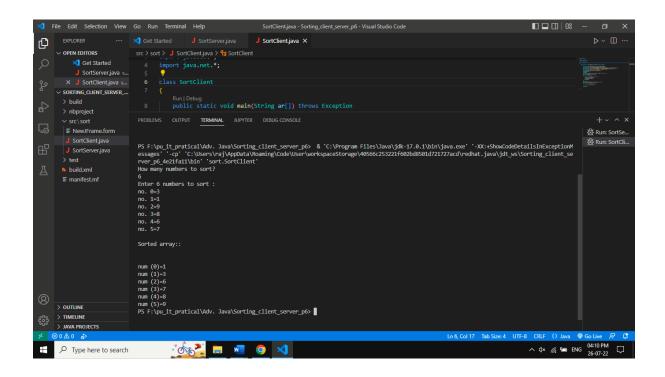
```
String sendarr=new String();
for (c = 0; c < n; c++)
{           sendarr+="\nnum ("+c+")="+arr[c];          }
System.out.println(sendarr);
p.println(sendarr);
p.flush();
s.close();
    }    }
```

Output:

PRACTICAL-7

Aim: Implement Student information system using JDBC and RMI.

Code:

```java
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.*;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JTextField;
import javax.swing.event.MouseInputListener;
import java.awt.event.*;
import java.sql.*;
import javax.swing.JOptionPane;
public class App extends JFrame implements ActionListener {
    JTextField jt1, jt2, jt3, jt4;
    App() {
        setSize(500, 600);
        setVisible(true);
        setLayout(null);
        JLabel l1, l2, l4;
        JButton jb1;
        jb1 = new JButton("Insert Value");
        jb1.setBounds(100, 150, 100, 40);
        jb1.addActionListener(this);
        add(jb1);
        l1 = new JLabel("Username   :");
        l2 = new JLabel("Password    :");
        jt1 = new JTextField();
        jt2 = new JTextField();
        l1.setBounds(50, 50, 200, 40);
        l2.setBounds(50, 100, 200, 40);
        add(l1);
        add(l2);
        jt1.setBounds(150, 50, 250, 40);
        jt2.setBounds(150, 100, 250, 40);
        add(jt1);
        add(jt2);    }
    public void actionPerformed(ActionEvent e) {
```
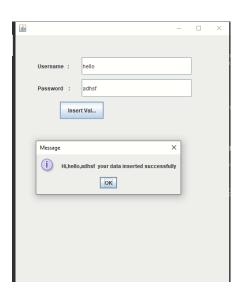
```
    String user1=jt1.getText();
    String pass1 = jt2.getText();
    String user = "root";
    String pass = "mysql";
    try {
        Class.forName("com.mysql.cj.jdbc.Driver");
        Connection con = DriverManager.getConnection(
"jdbc:mysql://localhost:3306/hemil", user, pass);
        String a = "insert into login1 values(?,?)";
        PreparedStatement ps = con.prepareStatement(a);
        ps.setString(1, user1);
        ps.setString(2, pass1);
        int rs=ps.executeUpdate();
        if(rs==1)
        {
        JOptionPane.showMessageDialog(this, "Hi,"+user1+","+pass1+"  your data inserted
successfully");
        System.out.println("Hemil ");
        }
        else{
            JOptionPane.showMessageDialog(this, "Hi,your data not inserted successfully");
        }
        con.close();
    } catch (Exception c) {          System.out.println(c);        }     }
  public static void main(String[] args) throws Exception {
    new App();            }        }
```

Output:

PRACTICAL-8

Aim: Call remote procedure from a jvm to another jvm by implementing RMI.

Code:

Client Program:
File: Client.java

```
import java.rmi.*;
import java.rmi.registry.*;
public class Client {
        public static void main(String args[])
        {       try
        {       Display d=(Display)Naming.lookup("rmi://localhost:1099/displayservice");
                System.out.println(d.dis());
                }catch(Exception e)
        {                       System.out.println(e);
        }       }       }
```

Defining the remote interface:
File: Display.java

```
import java.rmi.*;
public interface Display extends Remote
{
        public String dis() throws RemoteException;
}
```

Implementing the remote interface:
File: DisplayRemote.java

```
import java.rmi.server.*;
import java.rmi.*;
public class DisplayRemote extends UnicastRemoteObject implements Display{
        DisplayRemote()throws RemoteException
        {               super();        }
        public String dis()
        {return "hello World";}
}
```

Server Program:
File: Server.java

```
import java.rmi.*;

public class Server {
        public static void main(String args[])
```
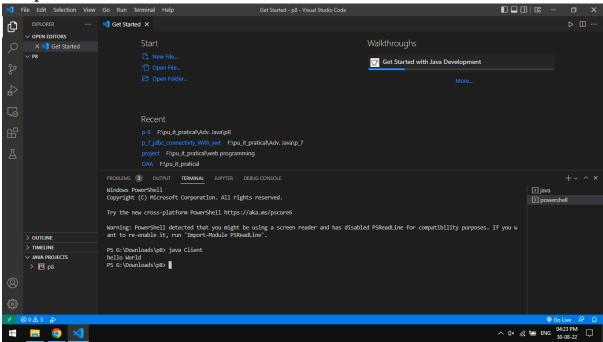
```
{
        try
        {
                Display d=new DisplayRemote();
                d.rebind("rmi://localhost:1099/displayService",d);
                }
        catch(Exception e)
        {System.out.println(e);
        }
        }
}
```

Output:

PRACTICAL-9

Aim: Make a simple calculator using RMI.

Code:

1. Addition interface:
File: AddInterface.java

```
import java.rmi.Remote;
import java.rmi.RemoteException;
public interface AddInterface extends Remote {
 public int Add(int x, int y) throws RemoteException;
}
```

2. Subtraction interface:
File: subInterface.java

```
import java.rmi.Remote;
import java.rmi.RemoteException;
public interface subInterface extends Remote {
 public int sub(int x, int y) throws RemoteException;
}
```

3. Multiplication interface:
File: mulInterface.java

```
import java.rmi.Remote;
import java.rmi.RemoteException;
public interface mulInterface extends Remote {
 public int mul(int x, int y) throws RemoteException;
}
```

4. Division interface:
File: divInterface.java

```
import java.rmi.Remote;
import java.rmi.RemoteException;
public interface divInterface extends Remote {
 public int div(int x, int y) throws RemoteException;
}
```

5. Implementation of the remote interface:
File: Impl.java

```
import java.rmi.server.*;
 public class Impl extends UnicastRemoteObject
 implements AddInterface, subInterface, mulInterface, divInterface {
  public Impl() throws Exception { super(); }
  public int Add(int x, int y) { return x + y; }
  public int sub(int x, int y) { return x - y; }
  public int mul(int x, int y) { return x * y; }
  public int div(int x, int y) { return x / y; }
```

```
}
```

6. Server program:
File:Server.java

```java
import java.rmi.*;
public class Server {
 public static void main(String[] args) throws Exception
 {
 Impl obj = new Impl();
 Naming.rebind("ADD", obj);
  System.out.println("Server Started");
 }
}
```

7. Client program:
File: Client.java

```java
import java.rmi.*;
import java.util.*;
public class Client {
   public static void main(String[] args) throws Exception {
      try (Scanner sc = new Scanner(System.in)) {
         while (true) {
      System.out.println("\n1.Addition\n2.Subtraction\n3.multiplication\n4.division\n5.Exit");
            System.out.println("Enter the option:");
            int opt = sc.nextInt();
            if (opt == 5) {
               break;            }
            System.out.println("Enter the the first number:");
            int a = sc.nextInt();
            System.out.println("Enter the second number:");
            int b = sc.nextInt();
            int n;
            switch (opt) {
               case 1:
                  AddInterface obj = (AddInterface) Naming.lookup("ADD");
                  n = obj.Add(a, b);
                  System.out.println("Addition= " + n);
                  break;
               case 2:
                  subInterface obj1 = (subInterface) Naming.lookup("ADD");
                  n = obj1.sub(a, b);
                  System.out.println("Subtraction= " + n);
                  break;
               case 3:
                  mulInterface obj2 = (mulInterface) Naming.lookup("ADD");
```

```
            n = obj2.mul(a, b);
            System.out.println("Multiplication = " + n);
            break;
        case 4:
        divInterface obj3 = (divInterface) Naming.lookup("ADD");
            n = obj3.div(a, b);
            System.out.println("Division = " + n);
            break;
        }       }     }   }    }
```

Output:

PRACTICAL-10

Aim: Study the functionalities of Eclipse/NetBeans and Connect to the Glassfish / Apache server.

Theory:

#NETBEANS:

NetBeans is an integrated development environment (IDE) for Java. NetBeans allows applications to be developed from a set of modular software components called modules. NetBeans runs on Windows, macOS, Linux and Solaris. In addition to Java development, it has extensions for other languages like PHP, C, C++, HTML5, and JavaScript. Applications based on NetBeans, including the NetBeans IDE, can be extended by third party developers.

Functionalities:

1. NetBeans IDE supports development of all Java application types (Java SE (including JavaFX), Java ME, web, EJB and mobile applications) out of the box.

2. Among other features are an Ant-based project system, Maven support, refactorings, version control (supporting CVS, Subversion, Git, Mercurial and Clearcase). Modularity: All the functions of the IDE are provided by modules. Each module provides a well-defined function, such as support for the Java language, editing, or support for the CVS versioning system, and SVN.

License: The IDE is licensed under the Apache License 2.0. Previously, from July 2006 through 2007, NetBeans IDE was licensed under Sun's Common Development and Distribution License (CDDL), a license based on the Mozilla Public License (MPL).

#APACHE:

1. Apache NetBeans is top level Apache Project dedicated to providing rock solid software development products (the Apache NetBeans IDE and the Apache NetBeans Platform) that address the needs of developers, users and the businesses who rely on NetBeans as a basis for their products; particularly, to enable them to develop these products quickly, efficiently and easily byleveraging the strengths of the Java platform and other relevant industry standards.

2. The two base products, the Apache NetBeans IDE and Apache NetBeans Platform, are free for commercial and non-commercial use, under the Apache license. The source code to both isavailable to anyone to reuse as they see fit, within the terms of use.

3. The Apache NetBeans project is also a vibrant community in which people from across the globe can ask questions, give advice, contribute and ultimately share in the success of our products. On 4. The NetBeans mailing lists and forums, you will find posts from students, developers from top companies, and individuals looking to expand their skills.

Functionalities:

1. Java. LSP/VS Code Integration. nb-javac. Gradle.

2. PHP. PHP 8.0 Support. Enhancements. Fixes. HTML & CSS.

3. C++ Lite.

4. Library Upgrades.

5. General Code Cleanup

PRACTICAL-11

Aim: Implement a simple Servlet application. Create directory structure, create references for web containers, create necessary web.xml and other config files and execute.

Code:

Index.html

```
<html>
<form action="LifeCycleServlet">
<input type="submit" value="invoke life cycle servlet">
</form>
</html>
```

Lifecycle.html

```
import javax.servlet.*;
import java.io.*;
public class LifeCycleServlet implements Servlet { ServletConfig config = null;
public void init(ServletConfig sc)
{           config = sc; System.out.println("in init");       }
public void service(ServletRequest req, ServletResponse res) throws ServletException,
IOException {
res.setContentType("text/html"); PrintWriter pw = res.getWriter();
pw.println("<h2>hello from life cycle servlet</h2>"); System.out.println("in service");
}
public void destroy()
{       System.out.println("in destroy");
} public String getServletInfo()
{           return "LifeCycleServlet";        }
public ServletConfig getServletConfig()
{           return config; // getServletConfig        }        }
```

Web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app>
<servlet>
<servlet-name>LifeCycleServlet</servlet-name>
<servlet-class>LifeCycleServlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>LifeCycleServlet</servlet-name>
<url-pattern>/LifeCycleServlet</url-pattern>
</servlet-mapping>
<session-config>
```
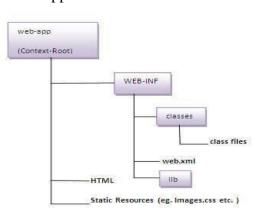
```
<session-timeout> 30
</session-config>
</web-app>
```


Output:

PRACTICAL-12

Aim: Create registration form of student using Servlet & JDBC.

Code:

First.java

```java
package servletdemo1;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
public class First extends HttpServlet {
 private static final long serialVersionUID = 1L;
 public First() {
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
 response.getWriter().append("Served at: ").append(request.getContextPath());
 PrintWriter pw=response.getWriter();
 String name=request.getParameter("name");
 String rollno=request.getParameter("rollno");
 String mobile_no=request.getParameter("mobile_no");
 try{kk obj= new kk();
 obj.demo(name,Integer.parseInt(rollno),Integer.parseInt(mobile_no));
 pw.println("record inserted successfully");
 }
catch(Exception e)
 pw.println(e.getMessage());
 }
 protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
 doGet(request, response); }}
```

kk.java

```java
package servletdemo1;
import java.beans.Statement;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
public class kk {
 public void demo(String name,int enroll,int mobileno) {
 try {Class.forName("oracle.jdbc.driver.OracleDriver");
Connection con =
```

```
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:orcl","system","mobile1");
PreparedStatement ps = con.prepareStatement("insert into student values(?,?,?)");
 ps.setString(1, name);
 ps.setInt(2, enroll);
 ps.setInt(3, mobileno);
 ps.executeUpdate();
 con.close();
 } catch (Exception e) {
 }}}
```

NewFile.html
```
<html>
<head><title>Insert title here</title>
<link href="bootstrap/css/bootstrap.min.css" rel="stylesheet" type="text/css" />
<script type="text/javascript" src="bootstrap/js/bootstrap.min.js"></script>
</head>
<body><div class="container">
<form role="form" action="First" method="get">
<div class="form-group">
<label>Student name</label>
<input type="text" class="form-control" name="name">
</div>
<div class="form-group">
<label>Enrollment number</label>
<input type="number" class="form-control" name="rollno">
</div>
<div class="form-group">
<label>Mobile no</label>
<input type="text" class="form-control" name="mobile_no">
</div>
<button type="submit" class="btn btn-default">Submit</button>
</form></div></body>
</html>
```

Output:

PRACTICAL-13

Aim: Create a JSP page that is a student registration form. Perform server side validations using JSP.

Code:

Std_reg.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE 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>Student_Registration</title>
   </head>   <body>
     <div class="container">
       <form role="form" action="validation" method="get">
         <div class="form-group">
           <label>Student name</label>
           <input type="text" class="form-control" name="name">
         </div>
         <div class="form-group">
           <label>Enrollment number</label>
           <input type="number" class="form-control" name="rollno">
         </div>
         <div class="form-group">
           <label>Mobile no</label>
           <input type="text" class="form-control" name="mobile_no">
         </div>
         <button type="submit" class="btn btn-default">Submit</button>
     </form>      </div>   </body></html>
```

validation.java

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet("/Validation")
public class validation extends HttpServlet {
 public validation() {           super();         }
```

```
protected void doGet(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
response.getWriter().append("Served at: ").append(request.getContextPath());
PrintWriter pw = response.getWriter();
String name = request.getParameter("name");
String rollno = request.getParameter("rollno");
String mobile_no=request.getParameter("mobile_no");
if(!(name.isEmpty() && rollno.isEmpty() && mobile_no.isEmpty()))
{ if (rollno.chars().allMatch( Character::isDigit ) &&
mobile_no.chars().allMatch(Character::isDigit ))
pw.println(" Valid Input");
else pw.println(" Enter numeric value in 2nd and 3rd textbox");
} else
pw.println(" Enter value in all the textboxs"); }
protected void doPost(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException { doGet(request, response); }
}
```

Output:

Student name [Hemil Chovatiya]
Enrollment number [200303108003]
Mobile no [7016824264]

[Submit]

Served at: /p13_jsp_reg_form Valid Input

PRACTICAL-14

Aim: Create a custom tag using JSP tag extension / library.

Code:
MyTagHandler.java

```java
package com.javatpoint.sonoo;
import java.util.Calendar;
import javax.servlet.jsp.JspException;
import javax.servlet.jsp.JspWriter;
import javax.servlet.jsp.tagext.TagSupport;
public class MyTagHandler extends TagSupport{
 public int doStartTag() throws JspException {
 JspWriter out=pageContext.getOut();//returns the instance of JspWriter
 try{ out.print(Calendar.getInstance().getTime());//printing date and time using JspWriter
 }catch(Exception e){System.out.println(e);}
 return SKIP_BODY;//will not evaluate the body content of the tag   }}
```

mytags.tld

```xml
<?xml version="1.0" encoding="ISO-8859-1" ?>
<!DOCTYPE taglib
 PUBLIC "-//Sun Microsystems, Inc.//DTD JSP Tag Library 1.2//EN"
 "http://java.sun.com/j2ee/dtd/web-jsptaglibrary_1_2.dtd">
<taglib> <tlib-version>1.0</tlib-version> <jsp-version>1.2</jsp-version>
 <short-name>simple</short-name>
 <uri>http://tomcat.apache.org/example-taglib</uri>
<tag><name>today</name>
<tag-class>com.javatpoint.sonoo.MyTagHandler</tag-class>
</tag></taglib>
```
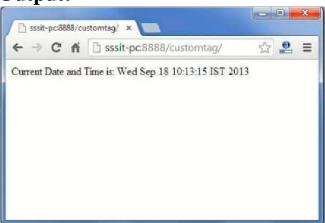
index.jsp

```jsp
<%@ taglib uri="WEB-INF/mytags.tld" prefix="m" %>
Current Date and Time is: <m:today/>
```

Output:

PRACTICAL-15

Aim: Create user interface of a student registration and login using JSF.

Code:

register.xhtml

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
 xmlns:h="http://java.sun.com/jsf/html"
 xmlns:f="http://java.sun.com/jsf/core">
<h:head>
 <title>Registration Page</title>
</h:head>
<h:body> <f:view>
 <h:form id="registerForm">
 <table> <tr>
 <td><h:outputText value="Enter Your First Name:" /></td>
 <td><h:inputText id="fname" value="#{user.firstName}"
 required="true" requiredMessage="Please enter your first name" /></td>
 <td><h:message for="fname" style="color:red" /></td>
 </tr> <tr>
 <td><h:outputText value="Enter Your Last Name:" /></td>
 <td><h:inputText id="lname" value="#{user.lastName}"
 required="true" requiredMessage="Please enter your last name" /></td>
 <td><h:message for="lname" style="color:red" /></td>
 </tr> <tr>
 <td><h:outputText value="Enter Your email ID:" /></td>
 <td><h:inputText id="email" value="#{user.email}"
required="true" requiredMessage="Please enter your email id" /></td>
 <td><h:message for="email" style="color:red" /></td>
 </tr> <tr>
 <td><h:outputText value="Enter Password :" /></td>
 <td><h:inputSecret id="psw" value="#{user.password}"
 required="true" requiredMessage="Please enter your password" /></td>
 <td><h:message for="psw" style="color:red" /></td>
 </tr> <tr>
 <td /> <td><h:commandButton value="Register" action="#{user.add}" /></td>
 </tr> <tr>
 <td><h:outputLink value="home.xhtml">Home</h:outputLink></td>
 </tr>
 </table> </h:form> </f:view></h:body></html>
```

success.xhtml

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<h:head>
 <title>Success Page</title>
</h:head><h:body> <f:view>
 <p>Successfully logged in</p>
<p>Hi, #{user.firstName}</p>
 <h:form> <p>
 <h:commandLink value="logout" action="#{user.logout}" />
 </p> </h:form> </f:view></h:body></html>
```

unsuccess.xhtml

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<h:head>
 <title>Unsuccess Page</title>
</h:head><h:body> <f:view>
 <p>There is an error in signing up. See Server Console for error.</p>
 <h:outputLink value="register.xhtml">Back</h:outputLink>
 </f:view></h:body></html>
```

User.java

```
package com.amzi.beans;
import java.sql.*;
public class User {
 private String firstName;
 private String lastName;
 private String email;
 private String password;
 private String dbPassword;
 private String dbName;
DataSource ds;
 public User() {
 try {    Context ctx = new InitialContext();
 ds = (DataSource) ctx.lookup("java:comp/env/jdbc/database");
 } catch (NamingException e) {
 e.printStackTrace();      }    }
 public String getDbPassword() {
 return dbPassword;    }
 public String getDbName() {
 return dbName;        }
 public String getFirstName() {
 return firstName;              }
 public void setFirstName(String name) {
```

```java
this.firstName = name;          }
public String getLastName() {
return lastName;          }
public String add() {
int i = 0;
if (firstName != null) {
PreparedStatement ps = null;
Connection con = null;
try {               if (ds != null) {
con = ds.getConnection();
if (con != null) {
String sql = "INSERT INTO user(firstname, password, lastname, email)
VALUES(?,?,?,?)";
ps = con.prepareStatement(sql);
ps.setString(1, firstName);
ps.setString(2, password);
ps.setString(3, lastName);
ps.setString(4, email);
i = ps.executeUpdate();
System.out.println("Data Added Successfully");      }          }
} catch (Exception e) {         System.out.println(e);           } finally {
try {    con.close();
ps.close();               } catch (Exception e) {
e.printStackTrace();  }    }   }
if (i > 0) {      return "success";        } else  return "unsuccess";      }
public void dbData(String uName) {
if (uName != null) {
PreparedStatement ps = null;
Connection con = null;
ResultSet rs = null;
if (ds != null) {
try {    con = ds.getConnection();
if (con != null) {
String sql = "select firstname,password from user where firstname = '" + uName + "'";
ps = con.prepareStatement(sql);
rs = ps.executeQuery();
rs.next();
dbName = rs.getString("firstname");
dbPassword = rs.getString("password");        }
} catch (SQLException sqle) {        sqle.printStackTrace();           }        }        }
public String login() {
dbData(firstName);
if (firstName.equals(dbName) && password.equals(dbPassword)) {
```

```
 return "output";
} else            return "invalid";            }
 public void logout() {
FacesContext.getCurrentInstance().getExternalContext()
.invalidateSession();
FacesContext.getCurrentInstance()
.getApplication().getNavigationHandler()
.handleNavigation(FacesContext.getCurrentInstance(), null, "/login.xhtml");        }        }
```
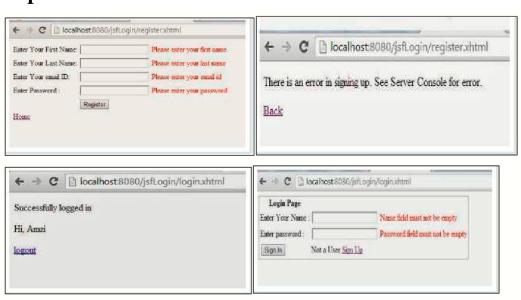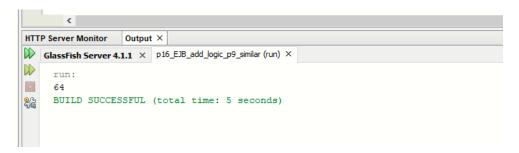
Output:

PRACTICAL-16

Aim: Transfer all the Business Logic to the EJB of practical 10.

Code:

AdderImplRemote.java
```
package com.javatpoint;
import javax.ejb.Remote;
@Remote
public interface AdderImplRemote {
 int add(int a, int b);
}
```

AdderImpl.java
```
package com.javatpoint;
import javax.ejb.Stateless;
@Stateless(mappedName = "st1")
public class AdderImpl implements AdderImplRemote {
 public int add(int a, int b) {
 return a + b;
 }
```

AdderImpl.java
```
 package com.javatpoint;
 import javax.naming.Context;
 import javax.naming.InitialContext;
 public class Test {
 public static void main(String[] args) throws Exception {
 Context context = new InitialContext();
 AdderImplRemote remote = (AdderImplRemote) context.lookup("st1");
 System.out.println(remote.add(32, 32)); } }
```

Output:

PRACTICAL-17

Aim: Create database and Implement JPA to provide persistence to practical 10.

Code:

```java
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.Table;
public class Employ {
 @Id
 @GeneratedValue(strategy = GenerationType.AUTO)
 private int id;
 private String name;
 private double sal;
 private String deg;
 public Employ(int id, String name, double sal, String deg) {
 super();
 this.id = id;
 this.name = name;
 this.sal = sal;
 this.deg = deg;
 }
 public Employ() {
 super();
 }
 public int getid() {
 return id;
 }
 public void setid(int eid) {
 this.id = id; }
 public String getname() {
 return name; }
 public void setname(String name) {
this.name = name; }
 public double getSal() {
 return sal; }
 public void setSal(double sal) {
 this.sal = sal; }
 public String getDeg() {
 return deg; }
 public void setDeg(String deg) {
 this.deg = deg; }
```

```java
@Override
public String toString() {
return "Employee [Id=" + id + ", Name=" + name + ", Salary=" + sal + ", deg=" + deg + "]";}}
```

Persist.xml <<? xml version = "1.0" encoding = "UTF-8" ? >

```xml
<persistence version = "2.0"
<properties >
<property name = "javax.persistence.jdbc.url"
value = "jdbc:mysql://localhost:3306/jpadb" / >
<property name = "javax.persistence.jdbc.user"
value = "root" / >
<property name = "javax.persistence.jdbc.password"
value = "root" / >
<property name = "javax.persistence.jdbc.driver"
value = "com.mysql.jdbc.Driver" / >
<property name = "eclipselink.logging.level"
value = "FINE" / >
```

CreateEmploy.java

```java
package mrbool.eclipselink.service;
import javax.persistence.EntityManager;
import javax.persistence.EntityManagerFactory;
import javax.persistence.Persist;
import eclipselink.entity.Employ;
public class CreateEmploy {
 public static void main(String[] args) {
 Employ employee = new Employ();
 employee.setid(101);
 employee.setname("Ravi");
 employee.setSalary(60000);
 employee.setDeg("Technical Support");
 entitymanager.persist(employee);
 entitymanager.getTransaction().commit();
 entitymanager.close();
 emfactory.close();
 }}
```

Output:

Id	Name	Salary	Deg
101	Ravi	60000	Technical Support