



S.I.E.S College of Arts, Science and Commerce
Sion(W), Mumbai – 400 022.

CERTIFICATE

This is to certify that Mr. **DeviPraveen Kumar Vengakeshu Naidu** Roll No. **SCS2324038**. Has successfully completed the necessary course of experiments in the subject of **Advanced Java [SIUSCS402]** during the academic year **2023 – 2024** complying with the requirements of **University of Mumbai**, for the course of **S.Y.BSc. Computer Science [Semester-4]**

Prof. In-Charge
Miss. Shivani Deopa
(Advanced Java)

Examination Date:
Examiner's Signature & Date:

Head of the Department
Prof.Manoj Singh

College Seal
And
Date

Sr. No.	Aim	Date	Signature
1	a. Write a Java Swing program to demonstrate basic Swing components JLabel, JTextField, JComboBox, JRadioButton, JButton etc) b. Write a Java Swing program to perform Login operations.		
2	Write a Java Swing program to demonstrate complex Swing components (JTable, JScrollPane, JMenu)		
3	Write a Java program to demonstrate Swing with JDBC performing database operations.		
4	Write a Java servlet to perform a. sum and product of two numbers b. calculate Net salary		
5	a. Write a Java application to demonstrate Servlet Life Cycle. b. Write a Java application to demonstrate Servlet Communication.		
6	Design database for user administration. Develop servlet(s) to perform CRUD operations.		
7	Create Employees table in EMP database. Perform select, insert, update, and delete operations on Employee table using JSP.		
8	Write a Student class with three properties(Name, Age, Standard). The useBean action declares a JavaBean for use in a JSP. Write Java application to access JavaBeans Properties.		
9	Write Java application to encoding and decoding JSON in Java.		

Practical 1

- a. Write a Java Swing program to demonstrate basic Swing components (JLabel, JTextField, JComboBox, JRadioButton, JButton etc) .

Code:

```
package pract1;

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

public class Pract1 extends JFrame implements ActionListener, ItemListener {

    JLabel l1, l2;
    JButton b1, b2;
    JTextField t1, t2;
    JComboBox jcb;
    String s = "";

    Pract1() {
        l1 = new JLabel("Enter a Number");
        l2 = new JLabel("Result : ");
        b1 = new JButton("Submit");
        b2 = new JButton("Clear");
        t1 = new JTextField(20);
        t2 = new JTextField(20);
        jcb = new JComboBox();
        jcb.addItem("Square");
        jcb.addItem("Cube");

        b1.addActionListener(this);
        b2.addActionListener(this);
        jcb.addItemListener(this);

        JPanel j1 = new JPanel();
        Container cp = getContentPane();
        j1.add(l1);
        j1.add(t1);
        j1.add(jcb);
        j1.add(b1);
        j1.add(b2);
        j1.add(l2);
        j1.add(t2);
        cp.add(j1);

        setSize(400, 350);
        setVisible(true);
    }
}
```

```

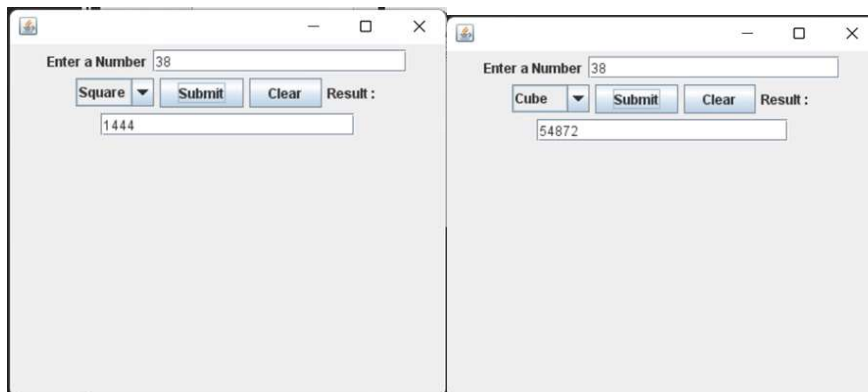
@Override
public void itemStateChanged(ItemEvent e) {
    s = (String) e.getItem();
}

@Override
public void actionPerformed(ActionEvent ae) {
    if (ae.getSource() == b1) {
        int n = Integer.parseInt(t1.getText());
        if (s.equals("Square")) {
            t2.setText(String.valueOf(n * n));
        } else if (s.equals("Cube")) {
            t2.setText(String.valueOf(n * n * n));
        }
    } else if (ae.getSource() == b2) {
        t1.setText("");
        t2.setText("");
        jcb.setSelectedIndex(0);
        s = "";
    }
}

public static void main(String[] args) {
    new Pract1();
}
}

```

Output:



b. Write a Java Swing program to perform Login operations.

Code:

```
package pract1;

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

public class pract1b extends JFrame implements ActionListener {

    JLabel l1, l2;
    JTextField t1, t2;
    JPasswordField psd;
    JButton bt;
    JPanel jp;

    pract1b() {

        l1 = new JLabel("Enter Username");
        l2 = new JLabel("Enter Password");
        t1 = new JTextField(20);
        psd = new JPasswordField(20);
        bt = new JButton("Login");
        jp = new JPanel(new GridLayout(3, 1));

        jp.add(l1);
        jp.add(t1);
        jp.add(l2);
        jp.add(psd);
        jp.add(bt);

        bt.addActionListener(this);
        add(jp, BorderLayout.CENTER);
        setTitle("Login Page");
        setSize(500, 200);
        setVisible(true);

    }

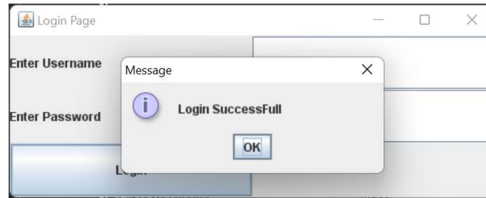
    @Override
    public void actionPerformed(ActionEvent ae) {
        if (ae.getSource() == bt) {
            String us = t1.getText();
            String ps = String.valueOf(psd.getPassword());
            if (us.equals("user") && ps.equals("pass")) {
                JOptionPane.showMessageDialog(rootPane, "Login Successful");
            } else {
                JOptionPane.showMessageDialog(rootPane, "Login Denied");
            }
        }
    }
}
```

```
}  
  
public static void main(String[] args) {  
    new pract1b();  
}  
}
```

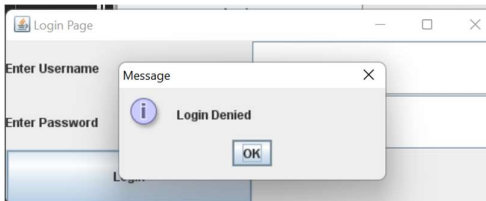
Output:



A screenshot of a Windows application window titled "Login Page". It contains two text input fields: "Enter Username" with the text "user" and "Enter Password" with masked characters "****". Below the fields is a blue "Login" button.



A screenshot of the "Login Page" window. The "Enter Username" field now contains the text "username". The "Enter Password" field still has "****". The "Login" button is at the bottom.



Practical 2

Write a Java Swing program to demonstrate complex Swing components (JTable, JScrollPane, JMenu)

Code:

```
package pract1;

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

class StudentDetails{
    JFrame sF;
    StudentDetails(){
        sF = new JFrame("Student Details");
        String data [][] = {
            {"Manohar", "101", "eg@gmail.com", "9876543210"},
            {"Amrita", "102", "eg@gmail.com", "9876534510"},
            {"Sheetal", "103", "eg@gmail.com", "9876543426"}
        };

        String column[]={"Name","RollNo","Email","PhoneNo"};
        JTable table = new JTable(data,column);
        table.setBounds(30,40,200,300);
        JScrollPane scrollPane = new JScrollPane(table);
        sF.add(scrollPane);
        sF.setSize(400,400);
        // sF.setLayout(null);
        sF.setVisible(true);
    }
}

class SQualifications{
    JFrame sqF;
    SQualifications(){
        sqF = new JFrame("Student Qualifications");
        String data [][] = {
            {"Manohar", "101", "12Fail"},
            {"Amrita", "102", "Garib", "9876534510"},
            {"Sheetal", "103", "Chor"}
        };

        String column[]={"Name","RollNo","Degrees"};
        JTable table = new JTable(data,column);
        table.setBounds(30,40,200,300);
        JScrollPane scrollPane = new JScrollPane(table);
        sqF.add(scrollPane);
        sqF.setSize(400,400);

        sqF.setVisible(true);
    }
}
```

```

    }
    class FeesBhara{
        JFrame fbF;
        FeesBhara(){
            fbF = new JFrame("Fees Bhara");
            String data [][] = {
                {"Mamohar", "101", "70000", "Pending"},
                {"Amrita", "102", "80000", "Garib"},
                {"Sheetal", "103", "30000", "Success"}
            };
            String column[]={"Name","RollNo","Fess","Status"};
            JTable table = new JTable(data,column);
            table.setBounds(30,40,200,300);
            JScrollPane scrollPane = new JScrollPane(table);
            fbF.add(scrollPane);
            fbF.setSize(400,400);

            fbF.setVisible(true);
        }
    }
}

```

```

public class pract2 extends JFrame implements ActionListener {
    JFrame mainFrame;
    JMenuBar mbar;
    JMenu Students, Teachers, Madat;
    JMenuItem StudentDetails,SQualifications,FeesBhara;

    pract2(){
        mainFrame = new JFrame("Menu");
        mbar = new JMenuBar();
        Students = new JMenu("Students");
        Teachers = new JMenu("Teachers");
        Madat = new JMenu("Madat");
        StudentDetails = new JMenuItem("Student Details");
        SQualifications = new JMenuItem("Student Qualifications");
        FeesBhara = new JMenuItem("Fees Bhara");
        StudentDetails.addActionListener(this);
        SQualifications.addActionListener(this);
        FeesBhara.addActionListener(this);

        Students.add(StudentDetails);
        Students.add(SQualifications);
        Students.add(FeesBhara);
        mbar.add(Students);
        mbar.add(Teachers);
        mbar.add(Madat);
        mainFrame.add(mbar);
        mainFrame.setJMenuBar(mbar);
        mainFrame.setLayout(null);
        mainFrame.setVisible(true);
        mainFrame.setSize(400,400);
    }
}

```



```

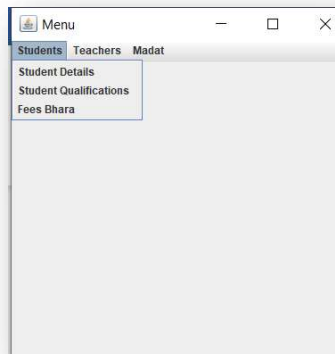
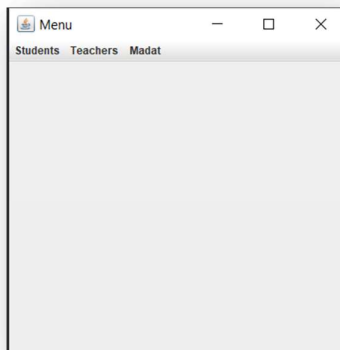
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        if(e.getSource()==StudentDetails){
            new StudentDetails();
        }
        if(e.getSource()==SQualifications){
            new SQualifications();
        }
        if(e.getSource()==FeesBhara){
            new FeesBhara();
        }
    }

    public static void main(String[] args) {
        new pract2();
    }
}

```

Output:



Student Details			
Name	RollNo	Email	PhoneNo
Manohar	101	eg@gmail.com	9876543210
Amrita	102	eg@gmail.com	9876543210
Sheetal	103	eg@gmail.com	9876543426

Student Qualifications		
Name	RollNo	Degrees
Manohar	101	12Fail
Amrita	102	Garib
Sheetal	103	Chor

Student Qualifications		
Name	RollNo	Degrees
Manohar	101	12Fail
Amrita	102	Garib
Sheetal	103	Chor

Practical 3

Write a Java program to demonstrate Swing with JDBC performing database operations.

Code:

```
package classroomadmin;

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

public class ClassroomAdmin extends JFrame implements ActionListener {

    JTextField t1, t2, t3, t4, t5, t6;
    JButton b1, b2, b3, b4;
    Container c;
    JList er;
    JPanel
studentLabelsPanel, studentButtonsPanel, studentManagementPanel, classManagementPanel;
    JLabel l1, l2, l3, l4, l5, l6;
    Connection con;
    Statement stmt;
    JTable table;
    JScrollPane scrollPane;

    public ClassroomAdmin() {
        setTitle("Classroom Administration");
        setSize(800, 600); // Increased frame size
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        c = getContentPane();
        c.setLayout(new BorderLayout());

        // Creating tabs for Student Management, Class Management, and
        // Timetable
        JTabbedPane tabbedPane = new JTabbedPane();
        studentManagementPanel = new JPanel();
        classManagementPanel = new JPanel();
        JPanel timetablePanel = new JPanel();
        tabbedPane.addTab("Student Management", studentManagementPanel);
        tabbedPane.addTab("Class Management", classManagementPanel);
        tabbedPane.addTab("Timetable", timetablePanel);
        c.add(tabbedPane, BorderLayout.CENTER);

        // Panel for labels and text fields in Student Management tab
        studentLabelsPanel = new JPanel(new GridLayout(3, 2));
        l1 = new JLabel("Roll No");
        l2 = new JLabel("Student Name");
        l3 = new JLabel("Address");
```

```

l4 = new JLabel("Telephone No.");
l5 = new JLabel("Course");
l6 = new JLabel("Elective Subject");
t1 = new JTextField(10);
t2 = new JTextField(10);
t3 = new JTextField(10);
t4 = new JTextField(10);
t5 = new JTextField(10);
t6 = new JTextField(10);

studentLabelsPanel.add(l1);
studentLabelsPanel.add(t1);
studentLabelsPanel.add(l2);
studentLabelsPanel.add(t2);
studentLabelsPanel.add(l3);
studentLabelsPanel.add(t3);
studentLabelsPanel.add(l4);
studentLabelsPanel.add(t4);
studentLabelsPanel.add(l5);
studentLabelsPanel.add(t5);
studentLabelsPanel.add(l6);
studentLabelsPanel.add(t6);

// Panel for buttons in Student Management tab
studentButtonsPanel = new JPanel(new FlowLayout());
b1 = new JButton("Add Student");
b2 = new JButton("Show All Students");
b3 = new JButton("Delete Student");
b4 = new JButton("Update Student Details");
studentButtonsPanel.add(b1);
studentButtonsPanel.add(b2);
studentButtonsPanel.add(b3);
studentButtonsPanel.add(b4);

studentManagementPanel.setLayout(new BorderLayout());
studentManagementPanel.add(studentLabelsPanel, BorderLayout.NORTH);
studentManagementPanel.add(studentButtonsPanel,
BorderLayout.SOUTH);

setVisible(true);

try {
    Class.forName("com.mysql.jdbc.Driver");
    con =
DriverManager.getConnection("jdbc:mysql://localhost:3308/ajsem4?zeroDateTi
meBehavior=convertToNull", "root", "");
    stmt = con.createStatement();
} catch (Exception ex) {
    JOptionPane.showMessageDialog(null, "Error : " + ex.getMessage(),
"Database Error", JOptionPane.ERROR_MESSAGE);
}

```

```

        // Action listeners for buttons
        b1.addActionListener(this);
        b2.addActionListener(this);
        b3.addActionListener(this);
        b4.addActionListener(this);
    }

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

    @Override
    public void actionPerformed(ActionEvent ae) {
        try {
            String rollno = t1.getText();
            String sname = t2.getText();
            String addr = t3.getText();
            String telno = t4.getText();
            String course = t5.getText();
            String elective_subject = t6.getText();

            if (ae.getSource() == b1) {
                // Insert operation
                String sql = "INSERT INTO students(rollno, name, address,
telephone, course, elective_subject) VALUES('\" + rollno + \"', '\" + sname + \"', '\"
+ addr + \"', '\" + telno + \"', '\" + course + \"', '\" + elective_subject + \"')";
                stmt.executeUpdate(sql);
                JOptionPane.showMessageDialog(null, "Record inserted");
            } else if (ae.getSource() == b2) {
                // Show all records
                ResultSet rs = stmt.executeQuery("SELECT * FROM students");
                ResultSetMetaData metaData = rs.getMetaData();

                int columns = metaData.getColumnCount();
                String[] columnNames = new String[columns];
                for (int i = 1; i <= columns; i++) {
                    columnNames[i - 1] = metaData.getColumnName(i);
                }

                Object[][] data = new Object[100][columns];
                int row = 0;
                while (rs.next()) {
                    for (int i = 1; i <= columns; i++) {
                        data[row][i - 1] = rs.getString(i);
                    }
                    row++;
                }

                // Remove previous table, if exists
                if (scrollPane != null) {
                    c.remove(scrollPane);
                }
            }
        }
    }

```

```

        // Add new table
        table = new JTable(data, columnNames);
        table.setEnabled(false);
        scrollPane = new JScrollPane(table);
        studentManagementPanel.add(scrollPane, BorderLayout.CENTER);
        validate();
    } else if (ae.getSource() == b3) {
        // Delete operation
        String deleteRollNo = JOptionPane.showInputDialog("Enter Roll No to
delete:");
        String deleteQuery = "DELETE FROM students WHERE rollno=" +
deleteRollNo + "";
        int deletedRows = stmt.executeUpdate(deleteQuery);
        if (deletedRows > 0) {
            JOptionPane.showMessageDialog(null, "Record deleted");
        } else {
            JOptionPane.showMessageDialog(null, "No record found for Roll
No: " + deleteRollNo);
        }
    } else if (ae.getSource() == b4) {
        // Update operation
        String updateRollNo = JOptionPane.showInputDialog("Enter Roll No
to update:");
        String updateQuery = "UPDATE students SET name=?, address=?,
telephone=?, course=?, elective_subject=? WHERE rollno=?";
        PreparedStatement pst = con.prepareStatement(updateQuery);
        pst.setString(1, sname);
        pst.setString(2, addr);
        pst.setString(3, telno);
        pst.setString(4, course);
        pst.setString(5, elective_subject);
        pst.setString(6, updateRollNo);
        int updatedRows = pst.executeUpdate();
        if (updatedRows > 0) {
            JOptionPane.showMessageDialog(null, "Record updated");
        } else {
            JOptionPane.showMessageDialog(null, "No record found for Roll
No: " + updateRollNo);
        }
    }
} catch (SQLException e) {
    JOptionPane.showMessageDialog(null, e.getMessage(), "Error",
JOptionPane.ERROR_MESSAGE);
}
}
}

```

Output:

The screenshot shows the 'Classroom Administration' application with the 'Student Management' tab selected. The form displays the following details:

- Roll No: 01
- Address: Chembur
- Course: CS
- Student Name: Praveen
- Telephone No.: 11111111
- Elective Subject: AJ

Below the form is a table with the following data:

rollno	name	address	telephone	course	elective_subject
1	Praveen	Chembur	11111111	CS	AJ

A confirmation dialog box is displayed with the message: "Record updated".

The screenshot shows the 'Classroom Administration' application with the 'Student Management' tab selected. The form displays the following details:

- Roll No: 01
- Address: Chembur
- Course: CS
- Student Name: Praveen
- Telephone No.: 11111111
- Elective Subject: AJ

Below the form is a table with the following data:

rollno	name	address	telephone	course	elective_subject
1	Praveen	Chembur	11111111	CS	AJ

An input dialog box is displayed with the message: "Enter Roll No to update:". The input field contains the value "1".

The screenshot shows the 'Classroom Administration' application with the 'Student Management' tab selected. The form displays the following details:

- Roll No: 01
- Address: Chembur
- Course: CS
- Student Name: Praveen
- Telephone No.: 11111111
- Elective Subject: AJ

Below the form is a table with the following data:

rollno	name	address	telephone	course	elective_subject
1	Praveen	Chembur	11111111	CS	AJ

A confirmation dialog box is displayed with the message: "Record updated".

rollno	name	address	telephone	course	elective_subject
1	Praveen	Chembur	11111111	CS	AJ

The screenshot shows the 'Classroom Administration' application with the 'Student Management' tab selected. The form displays the following details:

- Roll No: 01
- Address: Chembur
- Course: CS
- Student Name: Praveen
- Telephone No.: 11111111
- Elective Subject: AJ

Below the form is a table with the following data:

rollno	name	address	telephone	course	elective_subject
1	Praveen	Chembur	11111111	CS	AJ

An input dialog box is displayed with the message: "Enter Roll No to delete:". The input field contains the value "1".

The screenshot shows the 'Classroom Administration' application with the 'Student Management' tab selected. The form displays the following details:

- Roll No: 01
- Address: Chembur
- Course: CS
- Student Name: Praveen
- Telephone No.: 11111111
- Elective Subject: AJ

Below the form is a table with the following data:

rollno	name	address	telephone	course	elective_subject
1	Praveen	Chembur	11111111	CS	AJ

A confirmation dialog box is displayed with the message: "Record deleted".

rollno	name	address	telephone	course	elective_subject

Practical 4

Write a Java servlet to perform

- sum and product of two numbers.

Code:

Index.html

```
1 <!DOCTYPE html>
2 <!--
3 028
4 -->
5 <html>
6 <head>
7 <title>Servlet Demo</title>
8 <meta charset="UTF-8">
9 <meta name="viewport" content="width=device-width, initial-scale=1.0">
10 </head>
11 <body>
12 <div>
13 <form action="sumNProduct" method="post">
14 Enter First Number: <input type="number" name="num1"/><br/><br/>
15 Enter Second Number: <input type="number" name="num2"/><br/><br/>
16 <input type="submit" name="subbtn"/>
17 </form>
18 </div>
19 </body>
20 </html>
21
22
```

```
package pract4;
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;
public class sumNProduct extends HttpServlet {
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();

        try {
            double num1 = Double.parseDouble(request.getParameter("num1"));
            double num2 = Double.parseDouble(request.getParameter("num2"));
            double sum = num1 + num2;
            double product = num1 * num2;
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Calculated Result</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Calculated Result</h1>");
            out.println("<p>Sum: " + sum + "</p>");
            out.println("<p>Product: " + product + "</p>");
            out.println("</body>");
            out.println("</html>");
        } catch (NumberFormatException ex) {
            // Error handling if invalid input is provided
            out.println("<html>");
            out.println("<head>");
            out.println("<title>Error</title>");
            out.println("</head>");
            out.println("<body>");
            out.println("<h1>Error: Invalid input</h1>");
            out.println("<p>Please enter valid numbers.</p>");
            out.println("</body>");
            out.println("</html>");
        } finally {
            out.close();
        }
    }
}
```

- calculate Net salary

Code:

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Net Salary Calculator</title>
5  </head>
6  <body>
7      <h1>Net Salary Calculator</h1>
8      <form action="SalaryCal" method="post">
9          Enter Salary: <input type="text" name="salary"><br><br>
10         <input type="submit" value="Calculate Net Salary">
11     </form>
12 </body>
13 </html>
14
15
16

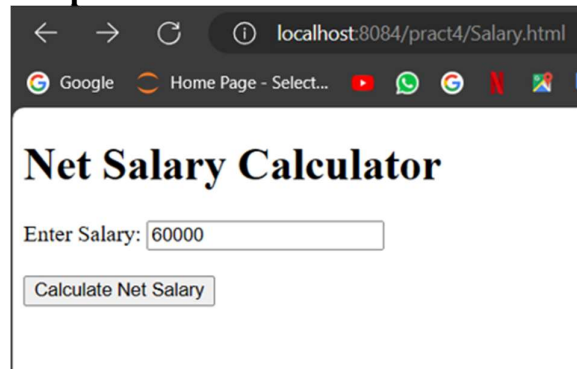
```

```

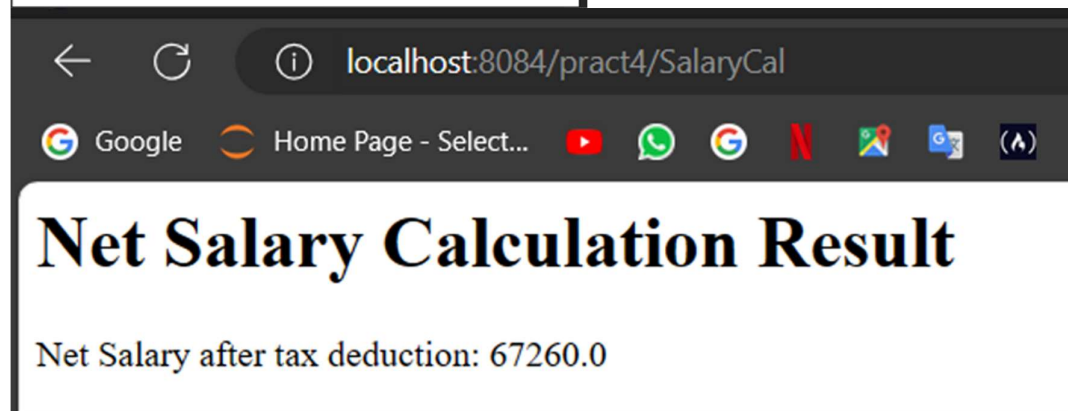
10 package pract4;
11 import java.io.IOException;
12 import java.io.PrintWriter;
13 import javax.servlet.ServletException;
14 import javax.servlet.http.HttpServlet;
15 import javax.servlet.http.HttpServletRequest;
16 import javax.servlet.http.HttpServletResponse;
17 public class SalaryCal extends HttpServlet{
18
19     protected void service(HttpServletRequest request, HttpServletResponse response)
20         throws ServletException, IOException {
21         response.setContentType("text/html;charset=UTF-8");
22         PrintWriter out = response.getWriter();
23
24         try {
25             // Get the values of the salary and tax rate from the request parameters
26             double salary = Double.parseDouble(request.getParameter("salary"));
27             double hra=0.10*salary;
28             double da=0.08*salary;
29             double grossSalary=salary+hra+da;
30             // Calculate net salary after tax deduction
31             double tax=0.05*grossSalary;
32             double netSalary=grossSalary-tax;
33             // Display the result
34             out.println("<html>");
35             out.println("<head>");
36             out.println("<title>Net Salary Calculation Result</title>");
37             out.println("</head>");
38             out.println("<body>");
39             out.println("<h1>Net Salary Calculation Result</h1>");
40             out.println("<p>Net Salary after tax deduction: " + netSalary + "</p>");
41             out.println("</body>");
42             out.println("</html>");
43         } catch (NumberFormatException ex) {

```


Output:



A screenshot of a web browser window. The address bar shows 'localhost:8084/pract4/Salary.html'. The browser's tab bar includes 'Google' and 'Home Page - Select...'. The main content area has a title 'Net Salary Calculator' in a large, bold, black serif font. Below the title is a text input field with the label 'Enter Salary:' and the value '60000'. Below the input field is a button labeled 'Calculate Net Salary'.



A screenshot of a web browser window. The address bar shows 'localhost:8084/pract4/SalaryCal'. The browser's tab bar includes 'Google', 'Home Page - Select...', and several other icons. The main content area has a title 'Net Salary Calculation Result' in a large, bold, black serif font. Below the title is a text line that reads 'Net Salary after tax deduction: 67260.0'.

Practical 5

a. Write a Java application to demonstrate Servlet Life Cycle.

Code:

```
7  -->
8  <html>
9  <head>
10     <title>Servlet LifeCycle Demoe</title>
11     <meta charset="UTF-8">
12     <meta name="viewport" content="width=device-width, initial-scale=1.0">
13 </head>
14 <body>
15     <form action="pract5a" method="get">
16         <input type="submit" value="Make Request"/>
17     </form>
18 </body>
19 </html>
20
```

```
package pract5a;
```

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

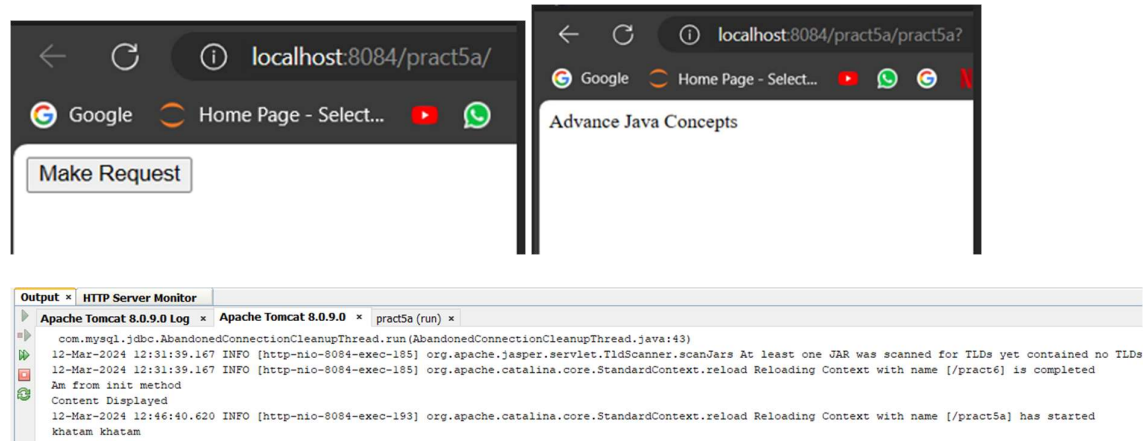
```
public class pract5a extends HttpServlet {
```

```
    private String output;
    public void init(ServletConfig config) throws ServletException{
        System.out.println("Am from init method");
        output="Advance Java Concepts";
    }
```

```
    public void doGet(HttpServletRequest req,HttpServletResponse resp) throws
ServletException,IOException{
        resp.setContentType("text/html");
        PrintWriter out = resp.getWriter();
        out.println(output);
        System.out.println("Content Displayed");
    }
```

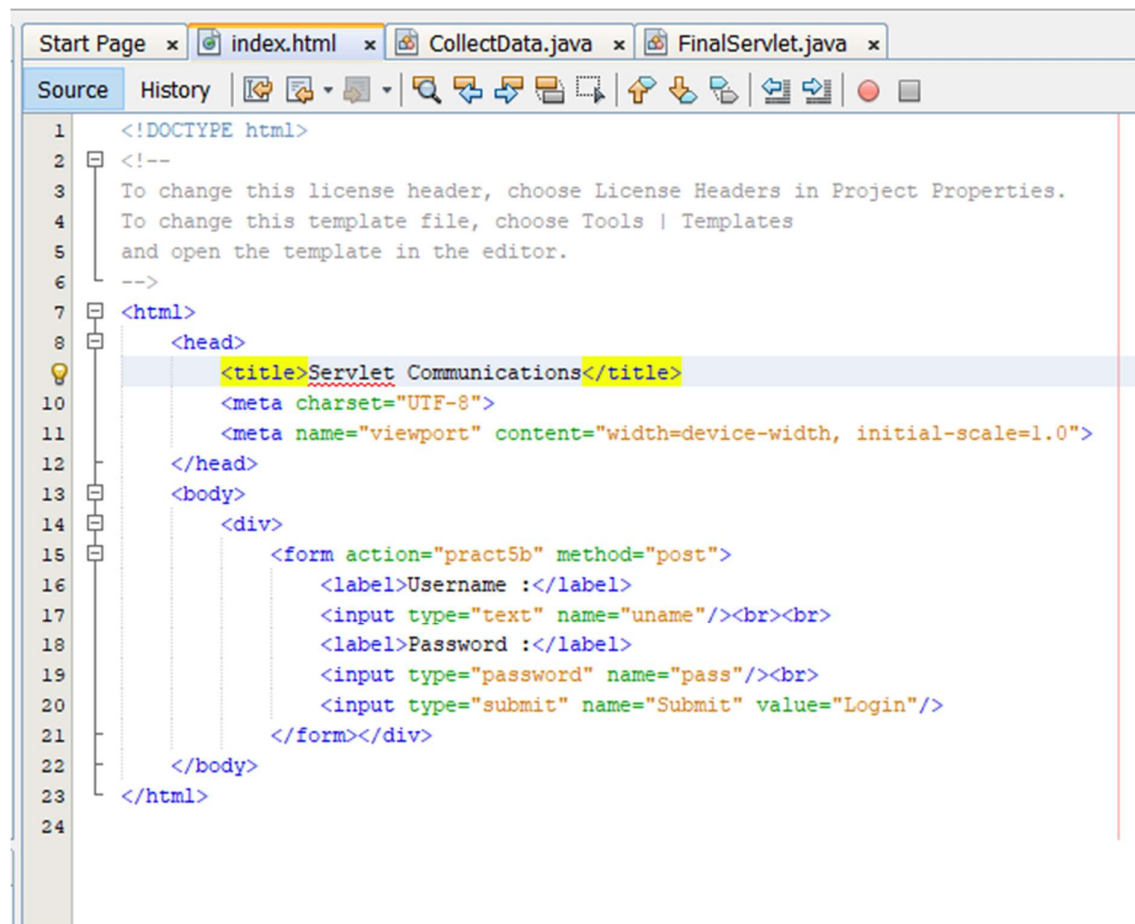
```
    public void destroy(){
        System.out.println("khatam khatam");
    }
}
```

Output:



b. Write a Java application to demonstrate Servlet Communication.

Code:



CollectData.java:

```
package pract5b;

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

public class CollectData extends HttpServlet {

    public void doPost(HttpServletRequest req,HttpServletResponse resp) throws
    ServletException,IOException{
        String uname = req.getParameter("uname");
        String pass = req.getParameter("pass");

        getServletContext().getRequestDispatcher("/FinalServlet").forward(req,resp);
    }
}
```

FinalServlet.java:

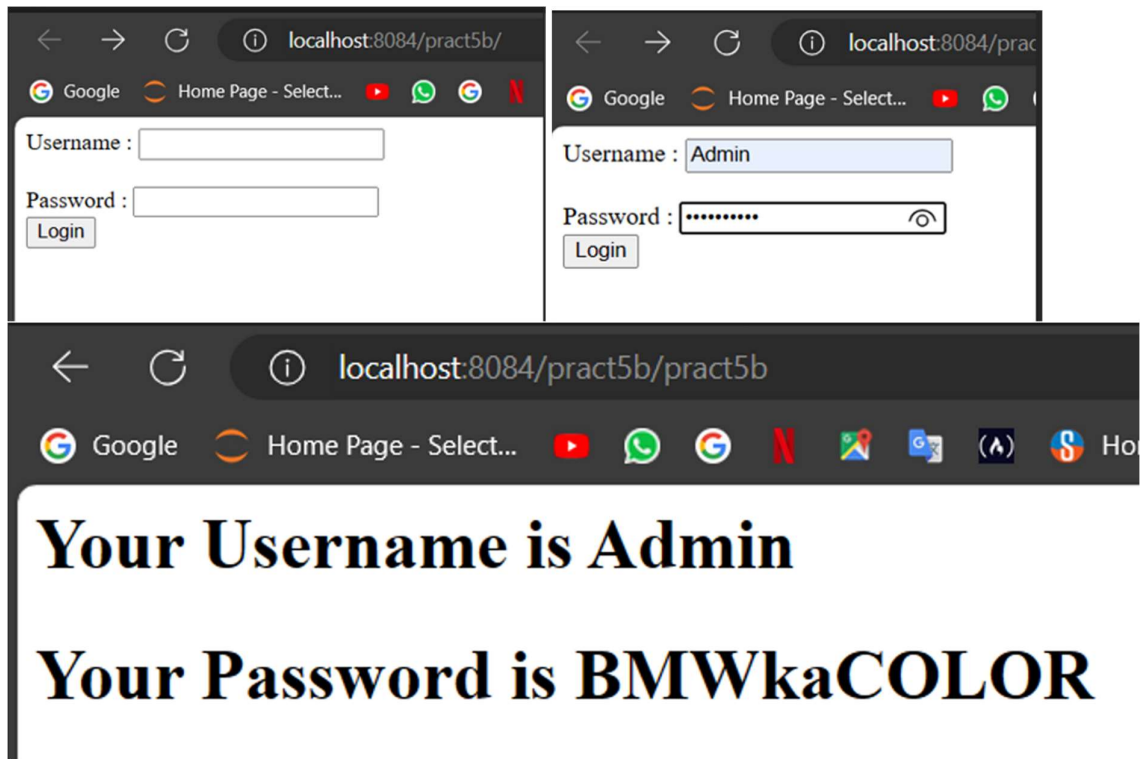
```
package pract5b;

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;

public class FinalServlet extends HttpServlet {

    public void doPost(HttpServletRequest req,HttpServletResponse resp) throws
    ServletException,IOException{
        String uname = req.getParameter("uname");
        String pass = req.getParameter("pass");
        PrintWriter pw = resp.getWriter();
        pw.println("<h1>Your Username is "+uname+"</h1>");
        pw.println("<h1>Your Password is "+pass+"</h1>");
    }
}
```

Output:



Practical 6

Design database for user administration. Develop servlet(s) to perform CRUD operations.

Code:

The Index page (Registration page)

```
<!DOCTYPE html>
<html>
  <head>
    <title>Register</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <style type="text/css"></style>
  </head>
  <body>
    <div>
      <form action="Adduser" method="post">
        Username: <input type="text" name="userName"/><br/><br/>
        Password: <input type="password" name="userPass"/><br/><br/>
        Email Id: <input type="text" name="userEmail"/><br/><br/>
        Country:
        <select name="userCountry">
          <option>India</option>
          <option>Bangladesh</option>
          <option>Srilanka</option>
          <option>Usa</option>
        </select>
        <br/>
        <br/>
        <input type="submit" value="Register"/>
        <br/>
        <br/>
        Already Registered,
        <a href="login.html" >Login Here</a>
      </form>
    </div>
  </body>
</html>
```

Adduser.Java (servlet):

```
package pract6;
import java.io.IOException;
import java.io.PrintWriter;
import java.sql.*;
import java.util.logging.Logger;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
public class Adduser extends HttpServlet {
    @Override
    public void doPost(HttpServletRequest req,HttpServletResponse resp)
    throws ServletException,IOException{
        resp.setContentType("text/html");
        String n=req.getParameter("userName");
        String p=req.getParameter("userPass");
        String e=req.getParameter("userEmail");
        String c=req.getParameter("userCountry");
        PrintWriter out = resp.getWriter();
        try{
            Class.forName("com.mysql.jdbc.Driver");
            Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3308/ajsem4?zeroDa
teTimeBehavior=convertToNull", "root", "");
            PreparedStatement ps=con.prepareStatement("insert into users
(username,password,email,country)values(?,?,?,?)");
            ps.setString(1, n);
            ps.setString(2, p);
            ps.setString(3, e);
            ps.setString(4, c);
            int i =ps.executeUpdate();
            if (i > 0) {
                out.print("You are successfully registered..<br/><br/>"
                + "Please <a href='login.html'>login here</a>.");
            } else {
                out.print("Registration failed.");
            }
        } catch (ClassNotFoundException | SQLException ex) {
            System.out.println(ex);
            out.print("An error occurred: " + ex.getMessage());
        } finally {
            out.close();
        }
    }
}
```

The Login page :

```
<html>
  <head>
    <title>Login Page</title>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
  </head>
  <body>
    <div>
      <form action="Getuser" method="post">
        Username: <input type="text" name="userName"/><br/><br/>
        Password: <input type="password" name="userPass"/><br/><br/>
        <input type="submit" value="Login"/>
      </form>
    </div>
  </body>
</html>
```

Getuser.Java (servlet)

```
package pract6;
```

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

```
public class Getuser extends HttpServlet {
```

```
    @Override
```

```
    public void doPost(HttpServletRequest req, HttpServletResponse resp)
        throws ServletException, IOException {
```

```
        resp.setContentType("text/html");
        String username = req.getParameter("userName");
        String password = req.getParameter("userPass");
        PrintWriter out = resp.getWriter();
        try {
```

```
            Class.forName("com.mysql.jdbc.Driver");
            Connection con =
```

```
DriverManager.getConnection("jdbc:mysql://localhost:3308/ajsem4?zeroDateTi
meBehavior=convertToNull", "root", "");
```

```
        PreparedStatement ps = con.prepareStatement("select * from users
where username=? and password=?");
```

```
        ps.setString(1, username);
        ps.setString(2, password);
        ResultSet rs = ps.executeQuery();
        if (rs.next()) {
```

```
            // User exists in the database, login successful
```

```
            out.print("<script>alert('Login successful. Welcome, " + username +
"!');");
```



```

        + "window.location='home.html?username=" + username +
        ""';</script>");

    } else {
        // User does not exist or credentials are incorrect
        out.print("<script>alert('Invalid username or password. Please try
again.');

```

The Home page

```

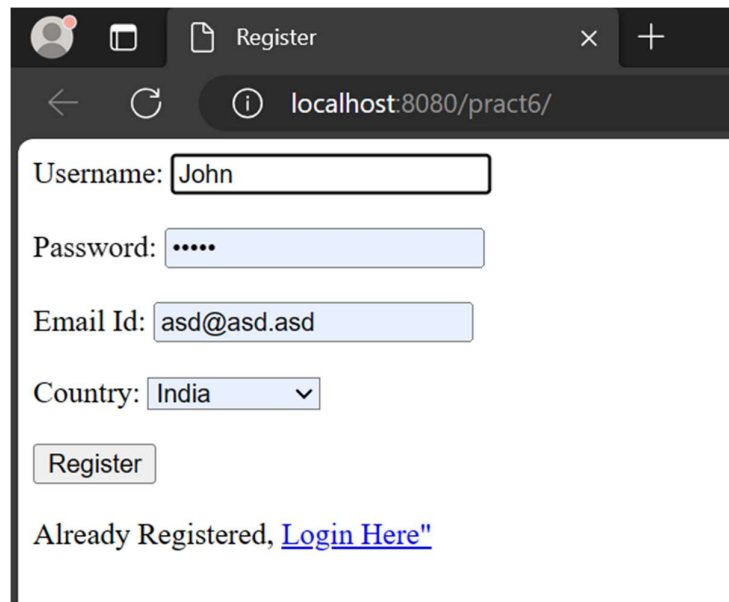
<!DOCTYPE html>
<html>
<head>
    <title>Home Page</title>
</head>
<body>
    <h1>Welcome to Your Home Page</h1>
    <p>Hello, <span id="username"></span>!</p>
    <br/>
    <form action="ChangePass">
        <input type="submit" value="Change Password"/>
    </form>
    <br/>
    <form action="DelUser">
        <input type="submit" value="Delete Account"/>
    </form>

    <script>
        // Function to get URL parameter by name
        function getUrlParameter(name) {
            name = name.replace(/\[/, '\\[').replace(/\]/, '\\]');
            var regex = new RegExp('[\\?&]' + name + '=(^&#*)');
            var results = regex.exec(location.search);
            return results === null ? '' : decodeURIComponent(results[1].replace(/\+/g, ' '));
        };

        // Get the username from the URL parameter
        var username = getUrlParameter('username');
        document.getElementById("username").innerText = username;
    </script>
</body>
</html>

```

Output:



Register

localhost:8080/pract6/

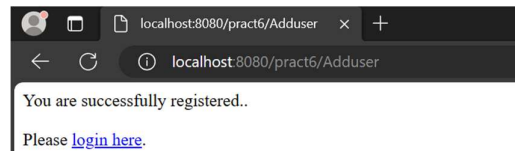
Username:

Password:

Email Id:

Country:

Already Registered, [Login Here](#)

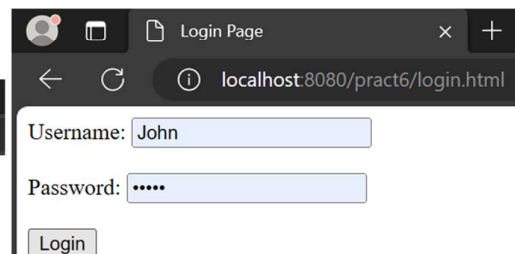


localhost:8080/pract6/Adduser

localhost:8080/pract6/Adduser

You are successfully registered..

Please [login here](#).

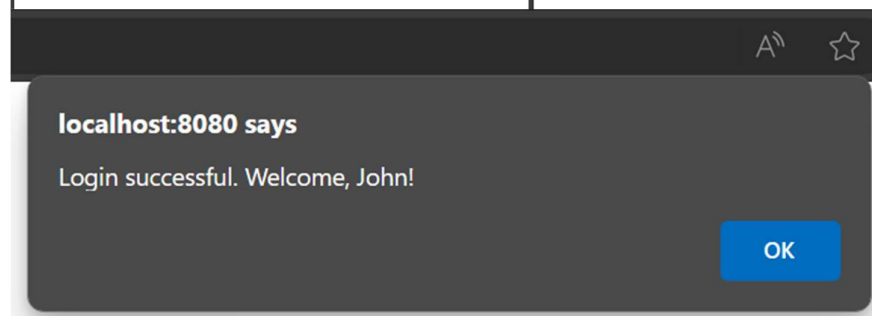


Login Page

localhost:8080/pract6/login.html

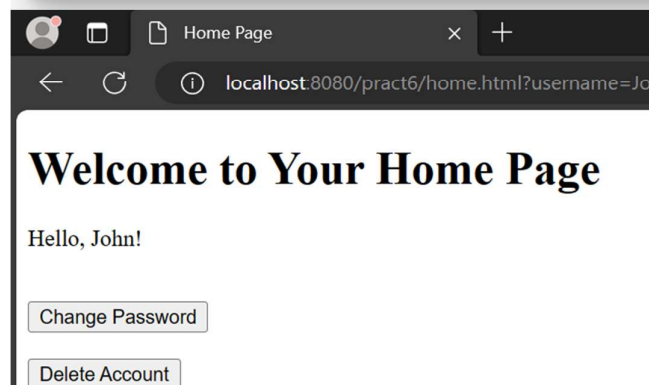
Username:

Password:



localhost:8080 says

Login successful. Welcome, John!



Home Page

localhost:8080/pract6/home.html?username=Jo

Welcome to Your Home Page

Hello, John!

Similarly for change password update servlet is used and for delete deluser

```
package pract6;
```

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

```
public class DelUser extends HttpServlet {
```

```
    @Override
```

```
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
```

```
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
```

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

```
        String confirm = request.getParameter("confirm");
```

```
        if (confirm != null && confirm.equals("Yes")) {
```

```
            deleteAccount(username, out, response);
```

```
        } else {
```

```
            response.sendRedirect("home.html?username=" + username);
```

```
        }
```

```
    }
```

```
    private void deleteAccount(String username, PrintWriter out,
    HttpServletResponse response) {
```

```
        try {
```

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

```
            try (Connection conn =
```

```
DriverManager.getConnection("jdbc:mysql://localhost:3308/ajsem4?zeroDateTi
meBehavior=convertToNull", "root", "")) {
```

```
                String sql = "DELETE FROM users WHERE username = ?";
```

```
                PreparedStatement pstmt = conn.prepareStatement(sql);
```

```
                pstmt.setString(1, username);
```

```
                int rowsDeleted = pstmt.executeUpdate();
```

```
                if (rowsDeleted > 0) {
```

```
                    out.println("<script>alert('Account deleted successfully!');"
```

```
                        + "window.location='login.html';</script>");
```

```
                } else {
```

```

        out.println("<script>alert('Failed to delete account. Please try
again.');"
        + "window.location='home.html?username='" + username +
        "';</script>");
    }

    } catch (ClassNotFoundException | SQLException ex) {
        out.println("An error occurred: " + ex.getMessage());
    } finally {
        out.close();
    }
}

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

```

Practical 7

Create Employees table in EMP database. Perform select, insert, update, and delete operations on Employee table using JSP.

Code:

```
<%@ page contentType="text/html" pageEncoding="UTF-8"%>
<%@ page import="java.io.*, java.util.*, java.sql.*"%>
<%@ page import="javax.servlet.http.*, javax.servlet.*"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql" %>

<!DOCTYPE html>
<html>
<head>
    <title>INSERT Operations</title>
</head>
<body>
    <sql:setDataSource var="jspdbc" driver="com.mysql.jdbc.Driver"
url="jdbc:mysql://localhost:3308/ajsem4?zeroDateTimeBehavior=convertToNull
" user="root" password="" />

    <!-- Insert Operation -->
    <sql:update dataSource="${jspdbc}" var="result1">
        INSERT INTO employee VALUES ('3','Joseph','Marketing','28','40000');
    </sql:update>

    <!-- Delete Operation -->
    <c:set var="empid" value="3" />
    <sql:update dataSource="${jspdbc}" var="del">
        DELETE FROM employee WHERE empid = ?;
        <sql:param value="${empid}" />
    </sql:update>

    <!-- Select Operation -->
    <sql:query dataSource="${jspdbc}" var="result">
        SELECT * FROM employee;
    </sql:query>

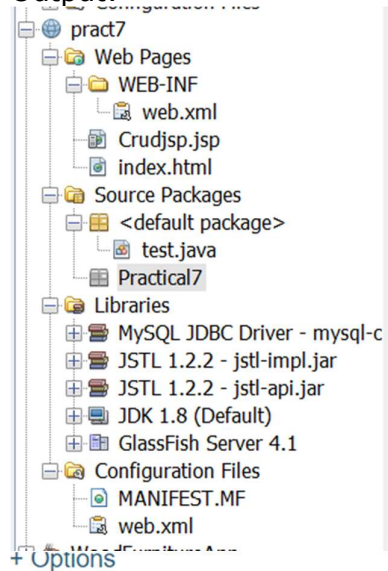
    <!-- Display the result in a table -->
    <table border="1" width="100%">
        <tr>
            <th>EMP Id</th>
            <th>EMP Name</th>
            <th>EMP Designator</th>
            <th>EMP Age</th>
            <th>EMP Salary</th>
        </tr>
        <c:forEach var="row" items="${result.rows}">
            <tr>
                <td><c:out value="${row.empid}" /></td>
```







```

        <td><c:out value="\${row.empname}" /></td>
        <td><c:out value="\${row.empdesg}" /></td>
        <td><c:out value="\${row.empage}" /></td>
        <td><c:out value="\${row.empsal}" /></td>
    </tr>
</c:forEach>
</table>
</body>
</html>

```

Output:



<div>← T →</div>				empid	empname	empdesg	empage	empsal
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	John	Manager	28	50000
<input type="checkbox"/>	 Edit	 Copy	 Delete	2	Joe	HR	28	45000

Check all With selected: Edit Copy Delete Export

EMP Id	EMP Name	EMP Designator	EMP Age	EMP Salary
1	John	Manager	28	50000
2	Joe	HR	28	45000

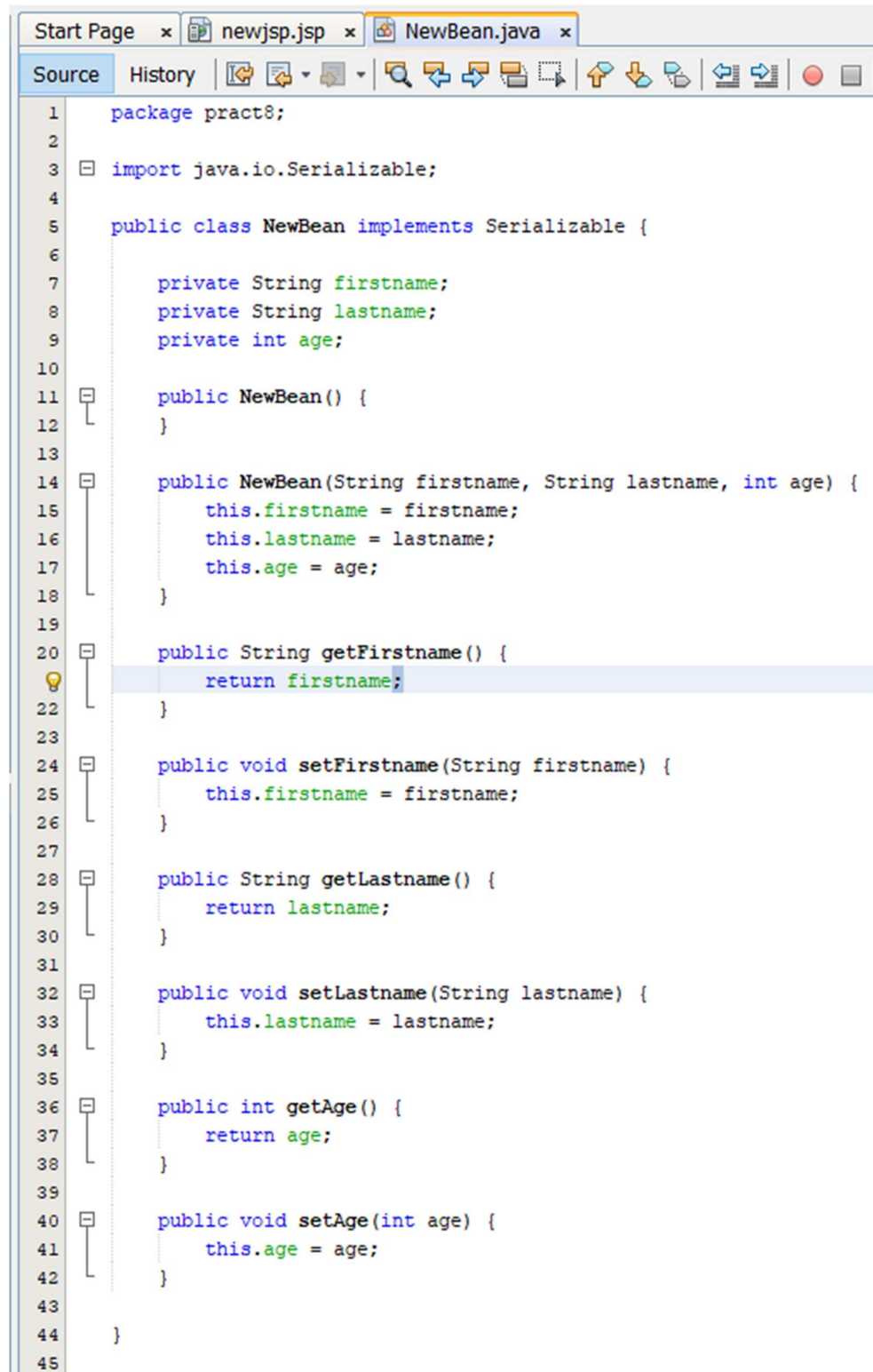
Practical 8

Write a Student class with three properties(Name, Age, Standard). The useBean action declares a JavaBean for use in a JSP. Write Java application to access JavaBeans Properties.

Code: JSP

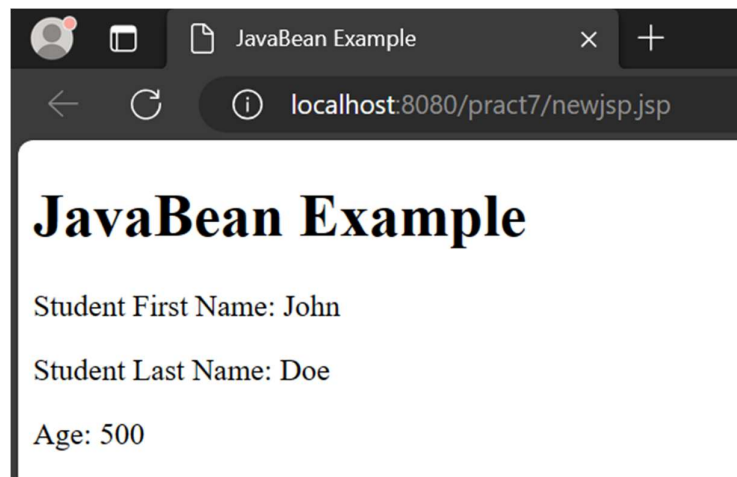
```
source History | 
1 <%@ page import="pract8.NewBean" %>
2 <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
3 <!DOCTYPE html>
4 <html>
5 <head>
6   <meta charset="UTF-8">
7   <title>JavaBean Example</title>
8 </head>
9 <body>
10  <h1>JavaBean Example</h1>
11  <jsp:useBean id="student" class="pract8.NewBean" />
12  <jsp:setProperty name="student" property="firstname" value="John" />
13  <jsp:setProperty name="student" property="lastname" value="Doe" />
14  <jsp:setProperty name="student" property="age" value="500" />
15  <p>Student First Name:
16    <jsp:getProperty name="student" property="firstname"/>
17  </p>
18  <p>Student Last Name:
19    <jsp:getProperty name="student" property="lastname"/>
20  </p>
21  <p>Age:
22    <jsp:getProperty name="student" property="age"/>
23  </p>
24 </body>
25 </html>
```

JavaBeans



```
1 package pract8;
2
3 import java.io.Serializable;
4
5 public class NewBean implements Serializable {
6
7     private String firstname;
8     private String lastname;
9     private int age;
10
11     public NewBean() {
12     }
13
14     public NewBean(String firstname, String lastname, int age) {
15         this.firstname = firstname;
16         this.lastname = lastname;
17         this.age = age;
18     }
19
20     public String getFirstname() {
21         return firstname;
22     }
23
24     public void setFirstname(String firstname) {
25         this.firstname = firstname;
26     }
27
28     public String getLastname() {
29         return lastname;
30     }
31
32     public void setLastname(String lastname) {
33         this.lastname = lastname;
34     }
35
36     public int getAge() {
37         return age;
38     }
39
40     public void setAge(int age) {
41         this.age = age;
42     }
43
44 }
45
```


Output:



Practical 9

Write Java application to encoding and decoding JSON in Java.

Code:

```
encoding
package pract9;
import java.io.FileNotFoundException;
import java.io. PrintWriter;
import java.util.LinkedHashMap;
import java.util.Map;
import org.json.simple.JSONArray;
import org.json.simple.JSONObject;

public class Pract9 {

    public static void main(String[] args) throws FileNotFoundException
    {
        JSONObject jo = new JSONObject();

        jo.put("firstname", "John");
        jo.put("lastname", "Smith");
        jo.put("age", 25);

        Map m=new LinkedHashMap (4);
        m.put("streetAddress", "21 2nd Street");
        m.put("city", "New York");
        m.put("state", "NY");
        m.put("postalCode", 10021);

        jo.put("address",m);

        JSONArray ja=new JSONArray();
        m=new LinkedHashMap (2);
        m.put("type", "home");
        m.put("number", "212-555-1234");
        ja.add(m);
        m=new LinkedHashMap(2);
        m.put("type", "fax");
        m.put("number", "212-555-1234");
        ja.add(m);
        jo.put("phonenumber", ja);
        PrintWriter pw = new PrintWriter("JSONExample.json");
        pw.write(jo.toJSONString());
        pw.flush();
        pw.close();
    }
}
```

Reader:

```
package pract9;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;
import java.util.Iterator;
import java.util.Map;
import org.json.simple.JSONArray;
import org.json.simple.JSONObject;
import org.json.simple.parser.*;

public class pract9b {
    public static void main(String args[]) throws Exception {
        try {
            Object obj = new JSONParser().parse(new FileReader
("JSONExample.json"));
            JSONObject jo = (JSONObject) obj;
            String firstName = (String) jo.get("firstname");
            String lastName = (String) jo.get("lastname");
            System.out.println("First Name: " + firstName);
            System.out.println("Last Name: " + lastName);

            long age = (long) jo.get("age");
            System.out.println("Age: " + age);

            Map address = (Map) jo.get("address");
            System.out.println("Address: ");
            Iterator<Map.Entry> itr1 = address.entrySet().iterator();
            while (itr1.hasNext()) {
                Map.Entry pair = itr1.next();
                System.out.println (pair.getKey() + " : " + pair.getValue());
            }

            JSONArray phoneNumbers = (JSONArray) jo.get("phonenumbers");
            System.out.println("Phone Numbers: ");
            Iterator itr2 = phoneNumbers.iterator();
            while (itr2.hasNext()) {
                System.out.println(itr2.next());
            }

        } catch (FileNotFoundException e) {
            System.out.println("File not found: " + e.getMessage());
        } catch (ParseException e) {
            System.out.println("Error parsing JSON file: " + e.getMessage());
        } catch (IOException e) {
            System.out.println("Error reading file: " + e.getMessage());
        }
    }
}
```

Output:

Writer:

event > Documents > NetBeansProjects > pract9

Name	Date modified	Type	Size
build	15-03-2024 00:46	File folder	
nbproject	15-03-2024 00:36	File folder	
src	15-03-2024 00:36	File folder	
build.xml	15-03-2024 00:36	Microsoft Edge HTM...	4 KB
JSONExample.json	15-03-2024 00:46	JSON Source File	1 KB
manifest.mf	15-03-2024 00:36	MF File	1 KB

```
1  {
2      "firstname": "John",
3      "address": {
4          "streetAddress": "21 2nd Street",
5          "city": "New York",
6          "state": "NY",
7          "postalCode": 10021
8      },
9      "phoneNumber": [
10         {
11             "type": "home",
12             "number": "212-555-1234"
13         },
14         {
15             "type": "fax",
16             "number": "212-555-1234"
17         }
18     ],
19     "age": 25,
20     "lastname": "Smith"
21 }
```

Reader:

Output

Java DB Database Process x GlassFish Server 4.1 x IDE Log x pract9 (run) x

```
run:
First Name: John
Last Name: Smith
Age: 25
Address:
streetAddress : 21 2nd Street
city : New York
postalCode : 10021
state : NY
Phone Numbers:
{"number":"212-555-1234","type":"home"}
{"number":"212-555-1234","type":"fax"}
BUILD SUCCESSFUL (total time: 0 seconds)
|
```