INDEX

S. NO.	PRACTICAL NAME	Page No.	Date
1.	WRITE A CODE FOR LISTS, TREES, TABLES, STYLED TEXT COMPONENTS, PROGRESS INDICATORS USING SWING.		
2.	WRITE A CODE FOR PROGRAMS USING TCP/IP CLIENT SOCKETS, TCP/IP SERVER SOCKETS		
3.	WRITE A CODE FOR PROGRAM WITH URL, URL CONNECTION AND DATA GRAMS CONNECTION.		
4.	WRITE A JDBC CODE FOR SELECT, INSERT, DELETE OPERATIONS ON A DATABASE.		
5.	WRITE A JDBC CODE FOR PREPARED STATEMENTS, SCROLLABLE AND UPDATABLE RESULT SETS, ROW SETS.		
6.	WRITE A SERVLET PROGRAM TO READ THE PARAMETERS.		
7.	WRITE A SERVLET CODE FOR USE OF COOKIES.		
8.	WRITE A CODE FOR SESSION TRACKING IN SERVLET.		
9.	WRITE A PROGRAM FOR CALLING ONE SERVLET BY ANOTHER.		
10.	WRITE A PROGRAM FOR USER VALIDATION USING JSP.		
11.	WRITE A PROGRAM FOR INSERTING DATA INTO TABLE USING JSP		
12.	WRITE A JAVA PROGRAM TO DEMONSTRATE THE CONCEPT OF BEAN		
13.	WRITE A JAVA PROGRAM TO DEMONSTRATE THE CONCEPT OF RMI		
14.	DESIGN HTML FORM FOR KEEPING STUDENT RECORD AND VALIDATE IT USING JAVA SCRIPT.		

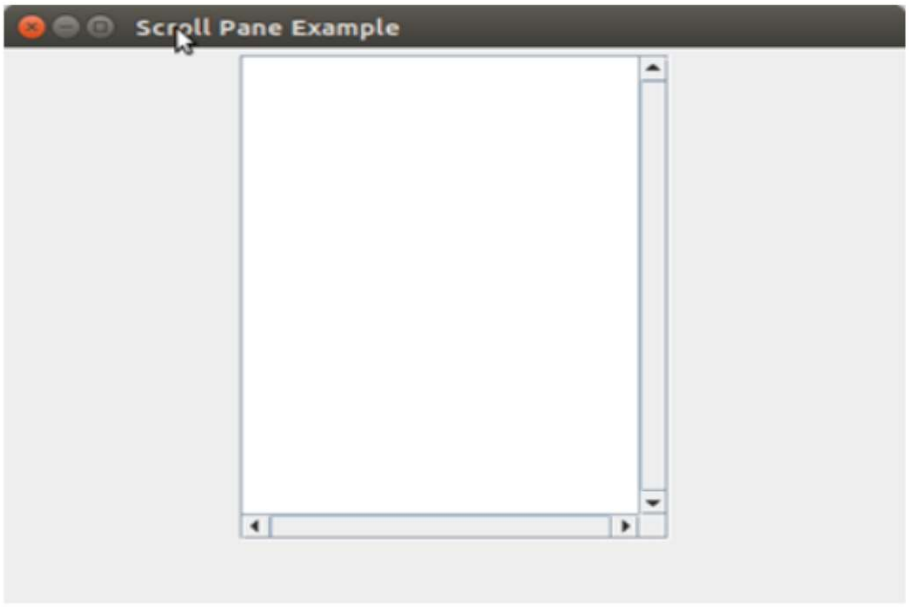
1. Write a code for lists, trees, table, styled text components, progress.

```
import java.awt.FlowLayout;
import javax.swing.JFrame;
import javax.swing.JScrollPane;
import javax.swing.JtextArea;
public class JScrollPaneExample {
   private static final long serialVersionUID = 1L;
   private static void createAndShowGUI() {
      // Create and set up the window.
      final JFrame frame = new JFrame("Scroll Pane Example");
      // Display the window.
      frame.setSize(500, 500);
      frame.setVisible(true);
      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      // set flow layout for the frame
      frame.getContentPane().setLayout(new FlowLayout());
      JTextArea textArea = new JTextArea(20, 20);
      JScrollPane scrollableTextArea = new JScrollPane(textArea);
      scrollableTextArea.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCROLLBAR_ALWAYS);
      scrollableTextArea.setVerticalScrollBarPolicy(JScrollPane.VERTICAL_SCROLLBAR_ALWAYS);
      frame.getContentPane().add(scrollableTextArea);
   }
   public static void main(String[] args) {
       javax.swing.SwingUtilities.invokeLater(newRunnable() {
           public void run() {
```

```
            createAndShowGUI();
        }
    });
  }
}
```

OUTPUT :

Output:



2. Write a code for programs using TCP/IP client sockets, TCP/IP server sockets.

```
import java.io.*;
import java.net.*;
class Client
{
public static void main(String data[])
{
//data is taken as command line argument
String ipAddress=data[0];
int portNumber=Integer.parseInt(data[1]);
int rollNumber=Integer.parseInt(data[2]);
String name=data[3];
String gender=data[4];
String request=rollNumber+”,”+name+”,”+gender+”#”;
//”#” acts as a terminator
try
{
Socket socket=new Socket(ipAddress , portNumber);
// Socket is initialized and attempt is made for connecting to the server
// Declaring other properties and streams
OutputStream outputStream;
OutputStreamWriter outputStreamWriter;
InputStream inputStream;
InputStreamReader inputStreamReader;
StringBuffer stringBuffer;
String response;
int x;
// retrieving output Stream and its writer, for sending request or acknowledgement
outputStream=socket.getOutputStream();
outputStreamWriter=new OutputStreamWriter(outputStream);
```

```
outputStreamWriter.write(request);
outputStreamWriter.flush();// request is sent
// retrieving input stream and its reader, for receiving    acknowledgement or response
inputStream=socket.getInputStream();
inputStreamReader=new InputStreamReader(inputStream);
stringBuffer=new StringBuffer();
while(true)
{
x=inputStreamReader.read();
if(x=='#' || x==-1) break; // reads till the terminator
stringBuffer.append((char)x);
}
response=stringBuffer.toString();
System.out.println(response);
socket.close(); //closing the connection
}catch(Exception exception)
{
// Raised in case, connection is refused or some other technical issue
System.out.println(exception);
}
}
}
```

Output –

Server side:

```
Socket successfully created..
Socket successfully binded..
Server listening..
server accept the client...
From client: hi
     To client : hello
From client: exit
     To client : exit
Server Exit...
```

Client side:

```
Socket successfully created..
connected to the server..
Enter the string : hi
From Server : hello
Enter the string : exit
From Server : exit
Client Exit...
```

3. Write a code for programs with URL, URL Connection and datagrams.

```
import java.io.*;
import java.net.*;
public class HttpURLConnectionDemo{
public static void main(String[] args){
try{
URL url=new URL("http://www.javatpoint.com/java-tutorial");
HttpURLConnection huc=(HttpURLConnection)url.openConnection();
for(int i=1;i<=8;i++){
System.out.println(huc.getHeaderFieldKey(i)+"= "+huc.getHeaderField(i));
}
huc.disconnect();
}catch(Exception e){System.out.println(e);}
}
}
```

OUTPUT :

Output:

```
Date = Thu, 22 Jul 2021 18:08:17 GMT
Server = Apache
Location = https://www.javatpoint.com/java-tutorial
Cache-Control = max-age=2592000
Expires = Sat, 21 Aug 2021 18:08:17 GMT
Content-Length = 248
Keep-Alive =timeout=5, max=1500
Connection = Keep-Alive
```

4. Write a JDBC code for SELECT, INSERT, DELETE operations on database.

INSERT:

```
// Java program to illustrate
// inserting to the Database
import java.sql.*;
public class insert1
{
	public static void main(String args[])
	{
		String id = "id1";
		String pwd = "pwd1";
		String fullname = "geeks for geeks";
		String email = "geeks@geeks.org";
		try
		{	Class.forName("oracle.jdbc.driver.OracleDriver");
			Connection con = DriverManager.getConnection("
			jdbc:oracle:thin:@localhost:1521:orcl", "login1", "pwd1");
			Statement stmt = con.createStatement();
			// Inserting data in database
			String q1 = "insert into userid values('" +id+ "', '" +pwd+
                                                    "', '" +fullname+ "', '" +email+ "')";
			int x = stmt.executeUpdate(q1);
			if (x > 0)
				System.out.println("Successfully Inserted");
			else	System.out.println("Insert Failed");
				con.close();
		}
		catch(Exception e){
			System.out.println(e);
		}
	}
}
```

UPDATE:

```
// Java program to illustrate
// updating the Database
import java.sql.*;
public class update1
{
	public static void main(String args[])
	{
		String id = "id1";
		String pwd = "pwd1";
		String newPwd = "newpwd";
		try
		{
			Class.forName("oracle.jdbc.driver.OracleDriver");
			Connection con = DriverManager.getConnection("
			jdbc:oracle:thin:@localhost:1521:orcl", "login1", "pwd1");
			Statement stmt = con.createStatement();
			// Updating database
			String q1 = "UPDATE userid set pwd = '" + newPwd +
					"' WHERE id = '" +id+ "' AND pwd = '" + pwd + "'";
			int x = stmt.executeUpdate(q1);
			if (x > 0)
				System.out.println("Password Successfully Updated");
			else
				System.out.println("ERROR OCCURED :(");
			con.close();
		}
		catch(Exception e)
		{
			System.out.println(e);
		}
	}
}
```

DELETE:

```
// Java program to illustrate
// deleting from Database
import java.sql.*;
public class delete
{
	public static void main(String args[])
	{
		String id = "id2";
		String pwd = "pwd2";
		try
		{
			Class.forName("oracle.jdbc.driver.OracleDriver");
			Connection con = DriverManager.getConnection("
			jdbc:oracle:thin:@localhost:1521:orcl", "login1", "pwd1");
			Statement stmt = con.createStatement();
			// Deleting from database
			String q1 = "DELETE from userid WHERE id = '" + id +
					"' AND pwd = '" + pwd + "'";
			int x = stmt.executeUpdate(q1);
			if (x > 0)
				System.out.println("One User Successfully Deleted");
			else
				System.out.println("ERROR OCCURED :(");
			con.close();
		}
		catch(Exception e)
		{
			System.out.println(e);
		}
	}
}
```

5. Write a JDBC code for Prepared Statements, Scrollable and Updatable Result Sets, Row sets.

```
import java.sql.*;
import java.io.*;
/**
 * This program demonstrates how to use scrollable result sets
 * that are sensitive to database changes with JDBC.
 * @author www.codejava.net
 */
public class ScrollableResultSetSensitiveExample {
    public static void main(String[] args) {
        String url = "jdbc:mysql://localhost:3306/college";
        String username = "root";
        String password = "password";
        Console console = System.console();
        try (Connection conn = DriverManager.getConnection(url, username, password)) {
            DatabaseMetaData metadata = conn.getMetaData();
            boolean isScrollSensitive =
metadata.supportsResultSetType(ResultSet.TYPE_SCROLL_SENSITIVE);
            if (!isScrollSensitive) {
                System.out.println("The database doesn't support scrollable and sensitive result sets.");
                return;
            }
            String sql = "SELECT * FROM student";
            Statement statement = conn.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
ResultSet.CONCUR_READ_ONLY);
```

```
        ResultSet result = statement.executeQuery(sql);
        int row = -1;
        while (row != 0) {
            row = Integer.parseInt(console.readLine("Enter row number: "));
            if (result.absolute(row)) {
                readStudentInfo("Student at row " + row + ": ", result);
            } else {
                System.out.println("There's no student at row " + row);
            }
        }
    } catch (SQLException ex) {
        ex.printStackTrace();
    }    }
private static void readStudentInfo(String position, ResultSet result) throws SQLException {
    String name = result.getString("name");
    String email = result.getString("email");
    String major = result.getString("major");
    String studentInfo = "%s: %s - %s - %s\n";
    System.out.format(studentInfo, position, name, email, major);
} }
```

6. Write a servlet program to read the parameters.

```
// Import required java libraries
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;

// Extend HttpServlet class
public class ReadParams extends HttpServlet {

  // Method to handle GET method request.
  public void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

    // Set response content type
    response.setContentType("text/html");

    PrintWriter out = response.getWriter();
    String title = "Reading All Form Parameters";
    String docType =
      "<!doctype html public \"-//w3c//dtd html 4.0 " + "transitional//en\">\n";

    out.println(docType +
      "<html>\n" +
      "<head><title>" + title + "</title></head>\n" +
      "<body bgcolor = \"#f0f0f0\">\n" +
      "<h1 align = \"center\">" + title + "</h1>\n" +
      "<table width = \"100%\" border = \"1\" align = \"center\">\n" +
      "<tr bgcolor = \"#949494\">\n" +
        "<th>Param Name</th>"
        "<th>Param Value(s)</th>\n"+
      "</tr>\n"
    );

    Enumeration paramNames = request.getParameterNames();

    while(paramNames.hasMoreElements()) {
      String paramName = (String)paramNames.nextElement();
      out.print("<tr><td>" + paramName + "</td>\n<td>");
      String[] paramValues = request.getParameterValues(paramName);

      // Read single valued data
      if (paramValues.length == 1) {
        String paramValue = paramValues[0];
        if (paramValue.length() == 0)
          out.println("<i>No Value</i>");
          else
          out.println(paramValue);
```

```
      } else {
        // Read multiple valued data
        out.println("<ul>");

        for(int i = 0; i < paramValues.length; i++) {
          out.println("<li>" + paramValues[i]);
        }
        out.println("</ul>");
      }
    }
    out.println("</tr>\n</table>\n</body></html>");
  }

  // Method to handle POST method request.
  public void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {

    doGet(request, response);
  }
}
```

Output:

Reading All Form Parameters

Param Name	Param Value(s)
maths	on
chemistry	on

7. Write a servlet code for use of Cookies

SERVLET 1:

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class FirstServlet extends HttpServlet {
  public void doPost(HttpServletRequestrequest, HttpServletResponseresponse){
    try{
    response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String n=request.getParameter("userName");
    out.print("Welcome "+n);
    Cookie ck=new Cookie("uname",n);//creating cookie object
    response.addCookie(ck);//adding cookie in the response
    //creating submit button
    out.print("<form action='servlet2'>");
    out.print("<input type='submit' value='go'>");
    out.print("</form>");
    out.close();
        }catch(Exception e){System.out.println(e);}
  }
}
```

SERVLET 2:

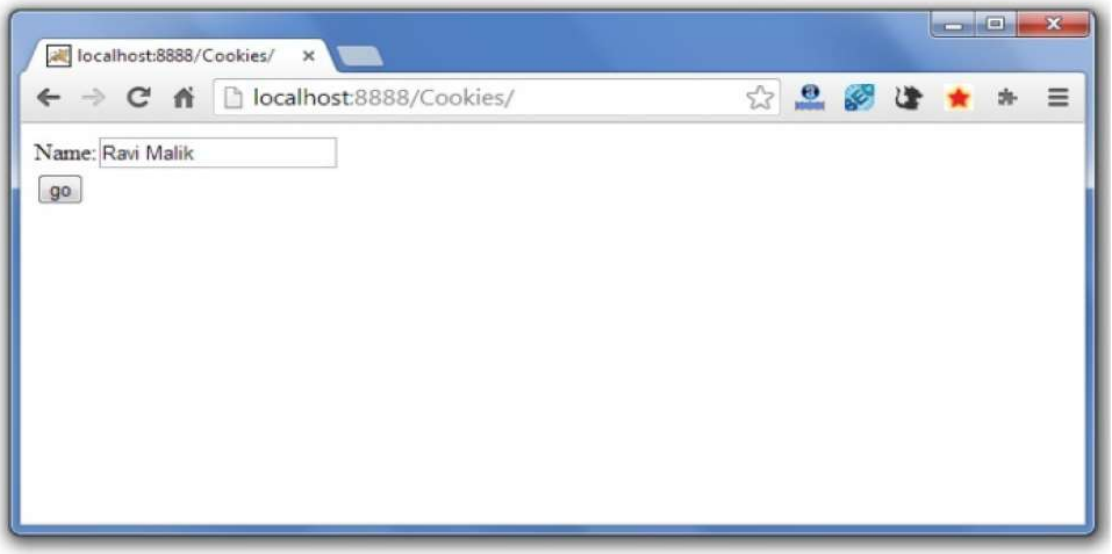
```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
```

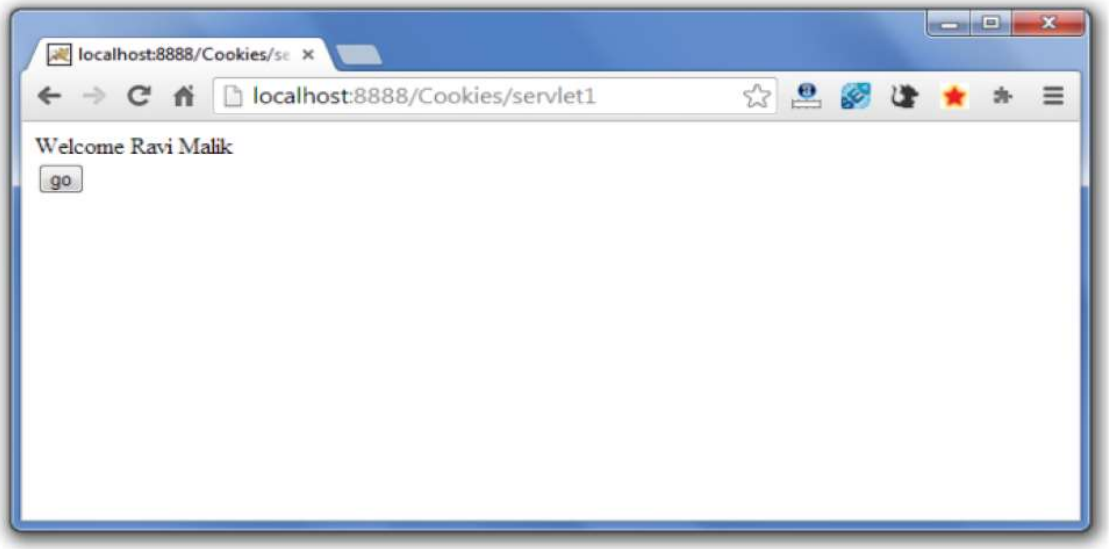
```
public class SecondServlet extends HttpServlet {
        public void doPost(HttpServletRequest request, HttpServletResponse response){
   try{
 response.setContentType("text/html");
   PrintWriter out = response.getWriter();
   Cookie ck[]=request.getCookies();
   out.print("Hello "+ck[0].getValue());
    out.close();
    }catch(Exception e){System.out.println(e);}
   }
 }
```

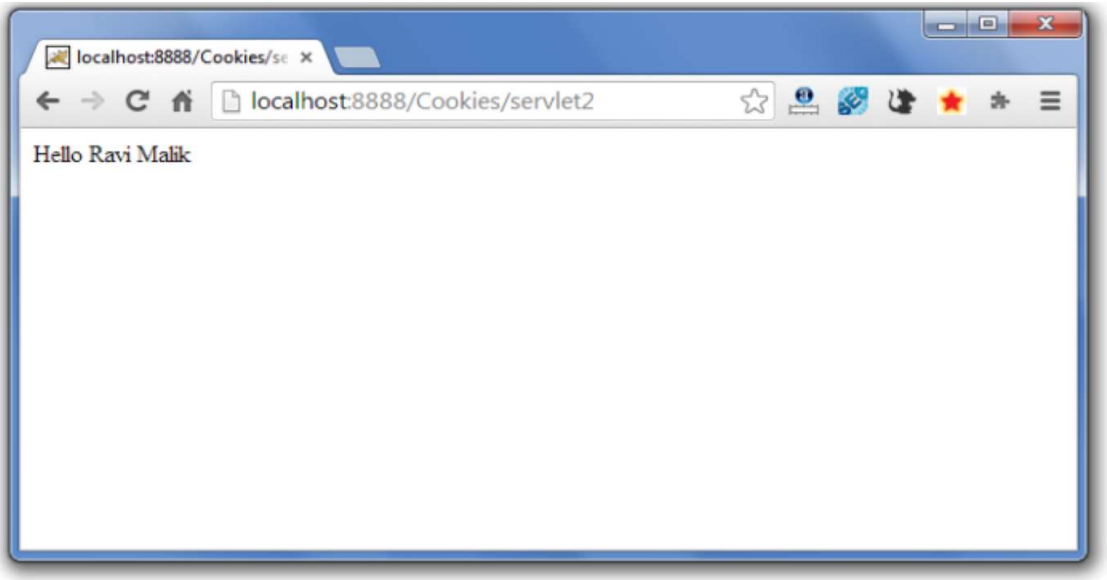
web.xml

```
<web-app>
<servlet>
<servlet-name>s1</servlet-name>
<servlet-class>FirstServlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>s1</servlet-name>
<url-pattern>/servlet1</url-pattern>
</servlet-mapping>
<servlet>
<servlet-name>s2</servlet-name>
<servlet-class>SecondServlet</servlet-class>
</servlet>
<servlet-mapping>
<servlet-name>s2</servlet-name>
<url-pattern>/servlet2</url-pattern>
</servlet-mapping>
</web-app>
```

Output:


localhost:8888/Cookies/
localhost:8888/Cookies/
Name: Ravi Malik
go


localhost:8888/Cookies/se
localhost:8888/Cookies/servlet1
Welcome Ravi Malik
go


localhost:8888/Cookies/se
localhost:8888/Cookies/servlet2
Hello Ravi Malik

8. Write a code for session tracking in servlet.

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
< div class
= "noIdeBtnDiv" > public class First extends HttpServlet {
   public void doGet(HttpServletRequest request, HttpServletResponse response) {
      try { /*Declaration of the get method*/
         response.setContentType("text/html"); // Setting the content type to text
         PrintWriter out = response.getWriter();
         String n = request.getParameter("userName"); /*Fetching the contents of the userName field
                                                        from the form*/
         out.print("Welcome " + n); // Printing the username
         HttpSession session = request.getSession(); /* Creating a new session*/
         session.setAttribute("uname", n);
         /*Setting a variable uname
                  containing the value as the fetched
                  username as an attribute of the session
                  which will be shared among different servlets of the application*/
         out.print("<a href='servlet2'>visit</a>"); // Link to the second servlet
         out.close();
      }
      catch (Exception e) {
         System.out.println(e);
      }
   }
}
```

9. Write a program for calling one servlet by another.

index.html

```
<form action="servlet1" method="post">
Name:<input type="text" name="userName"/><br/>
Password:<input type="password" name="userPass"/><br/>
<input type="submit" value="login"/>
</form>
```

Login.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
 public class Login extends HttpServlet {
 public void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
  response.setContentType("text/html");
    PrintWriter out = response.getWriter();
    String n=request.getParameter("userName");
    String p=request.getParameter("userPass");
    if(p.equals("servlet"){
        RequestDispatcher rd=request.getRequestDispatcher("servlet2");
        rd.forward(request, response);
    }
    else{
        out.print("Sorry UserName or Password Error!");
        RequestDispatcher rd=request.getRequestDispatcher("/index.html");
        rd.include(request, response);
        }
    }
 }
```

WelcomeServlet.java

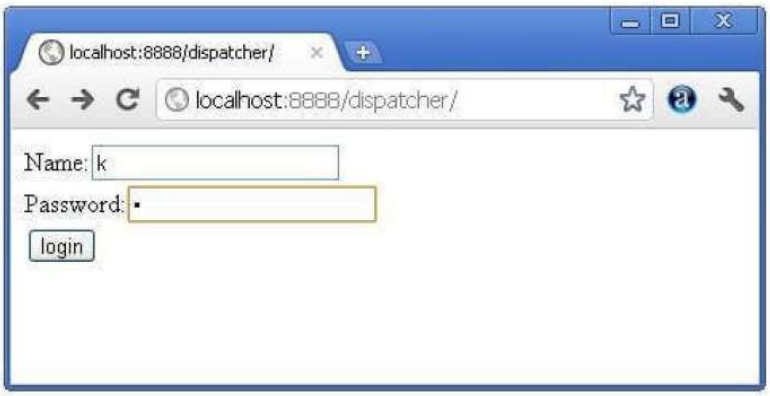
```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
```

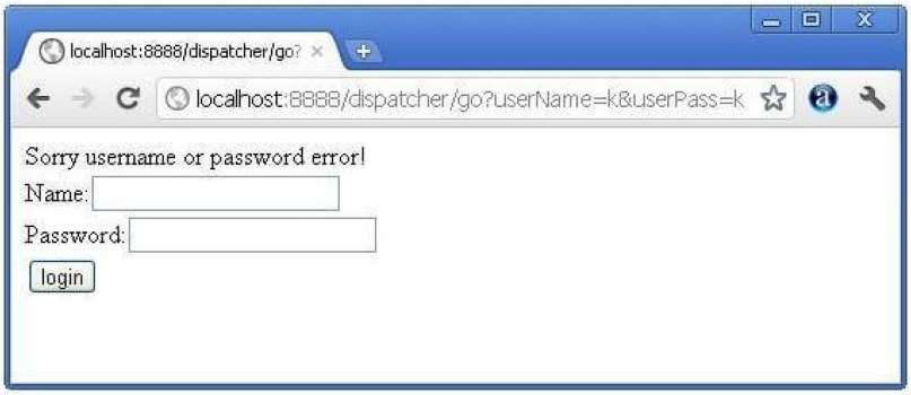
```
public class WelcomeServlet extends HttpServlet {
  public void doPost(HttpServletRequest request, HttpServletResponse response)
      throws ServletException, IOException {
   response.setContentType("text/html");
   PrintWriter out = response.getWriter();
   String n=request.getParameter("userName");
   out.print("Welcome "+n);
   }
}
```

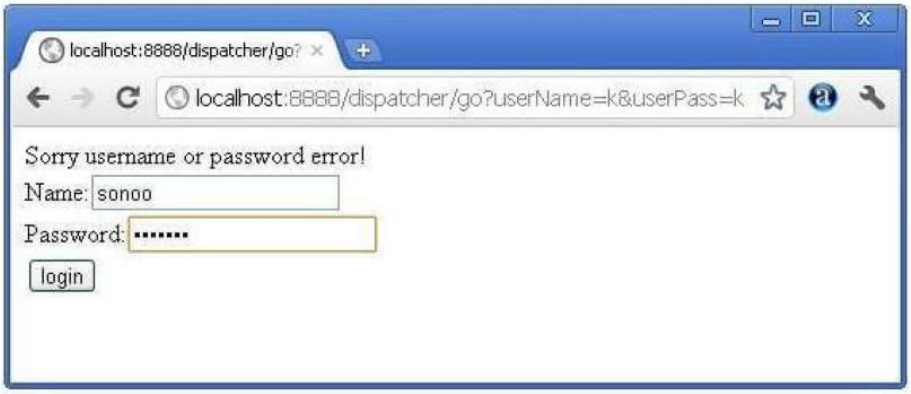
web.xml

```
<web-app>
 <servlet>
   <servlet-name>Login</servlet-name>
   <servlet-class>Login</servlet-class>
 </servlet>
 <servlet>
   <servlet-name>WelcomeServlet</servlet-name>
   <servlet-class>WelcomeServlet</servlet-class>
 </servlet>
 <servlet-mapping>
   <servlet-name>Login</servlet-name>
   <url-pattern>/servlet1</url-pattern>
 </servlet-mapping>
 <servlet-mapping>
   <servlet-name>WelcomeServlet</servlet-name>
   <url-pattern>/servlet2</url-pattern>
 </servlet-mapping>
 <welcome-file-list>
  <welcome-file>index.html</welcome-file>
 </welcome-file-list>
</web-app>
```

Output:


localhost:8888/dispatcher/
localhost:8888/dispatcher/
Name: k
Password:
login


localhost:8888/dispatcher/go?
localhost:8888/dispatcher/go?userName=k&userPass=k
Sorry username or password error!
Name:
Password:
login


localhost:8888/dispatcher/go?
localhost:8888/dispatcher/go?userName=k&userPass=k
Sorry username or password error!
Name: sonoo
Password:
login


localhost:8888/dispatcher/go?
localhost:8888/dispatcher/go?userName=sonoo&userPass=servlet
Welcome sonoo

10. Write a program for user validation using JSP

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>Login Page</title>
  </head>
  <body>
        <h1>User Details</h1>
        <%-- The form data will be passed to acceptuser.jsp
              for validation on clicking submit --%>
        <form method ="get" action="acceptuser.jsp">
              Enter Username : <input type="text" name="user"><br/><br/>
              Enter Password : <input type="password" name ="pass"><br/>
                    <input type ="submit" value="SUBMIT">
        </form>
  </body>
</html>
```

JSP to accept form data and verify a user : acceptuser.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  <title>Accept User Page</title>
  </head>
  <body>
        <h1>Verifying Details</h1>
        <%-- Include the ValidateUser.java class whose method
              boolean validate(String, String) we will be using --%>
```

```
<%-- Create and instantiate a bean and assign an id to
  uniquely identify the action element throughout the jsp --%>
        <jsp:useBean id="snr" class="saagnik.ValidateUser"/>
        <%-- Set the value of the created bean using form data --%>
        <jsp:setProperty name="snr" property="user"/>
        <jsp:setProperty name="snr" property="pass"/>
        <%-- Display the form data --%>
        The Details Entered Are as Under<br/>
        <p>Username : <jsp:getProperty name="snr" property="user"/></p>
        <p>Password : <jsp:getProperty name="snr" property="pass"/></p>
        <%-- Validate the user using the validate() of
                ValidateUser.java class --%>
        <%if(snr.validate("GeeksforGeeks", "GfG")){%>
                Welcome! You are a VALID USER<br/>
        <%}else{%>
                Error! You are an INVALID USER<br/>
        <%}%>
  </body>
</html>
```

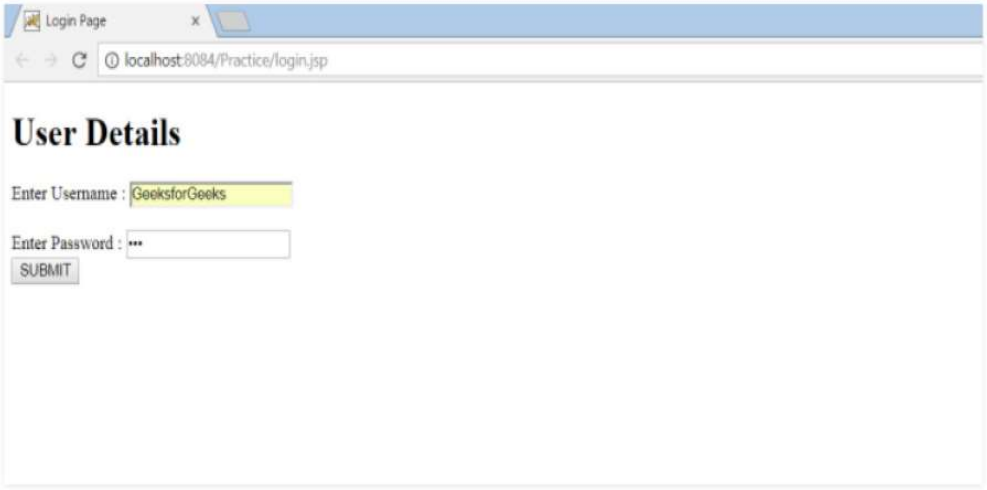
The ValidateUser.java class

```
package saagnik;
import java.io.Serializable;
// To persist the data for future use,
// implement serializable
public class ValidateUser implements Serializable {
  private String user, pass;
// Methods to set username and password
  // according to form data
  public void setUser(String u1) { this.user = u1; }
  public void setPass(String p1) { this.pass = p1; }
// Methods to obtain back the values set
  // by setter methods
  public String getUser() { return user; }
```

```
    public String getPass() { return pass; }
  // Method to validate a user
    public boolean validate(String u1, String p1)
    {
        if (u1.equals(user) && p1.equals(pass))
            return true;
        else
            return false;
    }
}
```

Output:

login.jsp



Next on clicking 'SUBMIT' button following page is generated.

acceptuser.jsp

11. Write a program for inserting data into table using JSP.

```
<%@ 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="gurucore"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="gurusql"%>
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Guru Database JSP1</title>
</head>
<body>

 <gurusql:setDataSource var="guru" driver="com.mysql.jdbc.Driver"
     url="jdbc:mysql://localhost/GuruTest"
     user="gururoot"  password="guru"/>

     <gurusql:update dataSource="${guru}" var="guruvar">
INSERT INTO guru_test VALUES (3, 'emp emp3');
</gurusql:update>
 </body>
</html>
```

12. Write a program to demonstrate the concept of bean.

```
/Employee.java
package mypack;
public class Employee implements java.io.Serializable{
private int id;
private String name;
public Employee(){}
public void setId(int id){this.id=id;}
public int getId(){return id;}
public void setName(String name){this.name=name;}
public String getName(){return name;}
}
```

How to access the JavaBean class?

To access the JavaBean class, we should use getter and setter methods.

```
package mypack;
public class Test{
public static void main(String args[]){
Employee e=new Employee();//object is created
e.setName("Arjun");//setting value to the object
System.out.println(e.getName());
}}
```

13. Write a program to demonstrate the concept of RMI.

Program: Power.java

```
import java.rmi.*;
public interface Power extends Remote {
  public int power1()throwsRemoteException;
}
```

Program: PowerRemote.java

```
import java.rmi.*;
import java.rmi.server.*;
import java.util.Scanner;
public class PowerRemote extends UnicastRemoteObject implements Power {
PowerRemote()throws RemoteException {
  super();
}
public int power1(int z) {
int z;
Scanner sc = new Scanner(System.in);
System.out.println("Enter the base number ::");
int x = sc.nextInt();
System.out.println("Enter the exponent number ::");
int y = sc.nextInt();
            z=y^x;
System.out.println(z);
}
}
```

MyServer.java

```
import java.rmi.*;
import java.rmi.registry.*;
public class MyServer
{
public static void main(String args[])
```

```
{
try
{
Power stub=new PowerRemote();
Naming.rebind("rmi://localhost:1995/shristee",stub);
}
catch(Exception e)
{
System.out.println(e);
}
}
}
```

MyClient.java

```
import java.rmi.*;
public class MyClient
{
public static void main(String args[])
{
try
{
Power stub=(Power)Naming.lookup("rmi://localhost:1995/shristee");
System.out.println(stub.power1());
}
catch(Exception e){}
}
}
```

Output:


E:\java\rmi>javac *.java
E:\java\rmi>rmic PowerRemote
E:\java\rmi>rmiregistry 5000
E:\>cd java
E:\java>cd rmi
E:\java\rmi>java MyServer
E:\>cd java
E:\java>cd rmi
E:\java\rmi>java MyClient
256.0

14. Design a HTML form for keeping student record and validate it using JAVASCRIPT.

```
<body>
    <h1 style="text-align: center;">REGISTRATION FORM</h1>
    <form name="RegForm" action="/submit.php"
        onsubmit="return GEEKFORGEEKS()" method="post">
        <p>Name: <input type="text" size="65" name="Name" /></p> <br />
        <p>Address: <input type="text" size="65" name="Address" /></p> <br />
        <p>E-mail Address: <input type="text" size="65" name="EMail" /></p> <br />
        <p>Password: <input type="text" size="65" name="Password" /></p> <br />
        <p>Telephone: <input type="text" size="65" name="Telephone"/></p><br/>
        <p>
            SELECT YOUR COURSE
            <select type="text" value="" name="Subject">
                <option>BTECH</option>
                <option>BBA</option>
                <option>BCA</option>
                <option>B.COM</option>
                <option>GEEKFORGEEKS</option>
            </select> </p>
        <br /> <br />
        <p>Comments: <textarea cols="55" name="Comment"> </textarea></p>
        <p>
            <input type="submit" value="send" name="Submit" />
            <input type="reset" value="Reset" name="Reset" />
        </p>
    </form>
</body>
<script>
    function GEEKFORGEEKS() {
        var name = document.forms["RegForm"]["Name"];
        var email = document.forms["RegForm"]["EMail"];
        var phone = document.forms["RegForm"]["Telephone"];
```

```
var what = document.forms["RegForm"]["Subject"];
var password = document.forms["RegForm"]["Password"];
var address = document.forms["RegForm"]["Address"];
if (name.value == "") {
        window.alert("Please enter your name.");
        name.focus();
        return false;
}
if (address.value == "") {
        window.alert("Please enter your address.");
        address.focus();
        return false;
}
if (email.value == "") {
        window.alert(
        "Please enter a valid e-mail address.");
        email.focus();
        return false;
}
if (phone.value == "") {
        window.alert(
        "Please enter your telephone number.");
        phone.focus();
        return false;
}
if (password.value == "") {
        window.alert("Please enter your password");
        password.focus();
        return false; }
if (what.selectedIndex < 1) {
        alert("Please enter your course.");
        what.focus();
        return false; }
```

```
                return true;
        }
</script>
```

Styling the form:

```
<style>
div {
        box-sizing: border-box;
        width: 100%;
        border: 100px solid black;
        float: left;
        align-content: center;
        align-items: center;
}
form {
        margin: 0 auto;
        width: 600px;
}</style>
```

COMBINED CODE [ALL OF THE ABOVE SECTIONS CLUBBED)

```
<html>
        <head>
                <script>
                        function GEEKFORGEEKS() {
                                var name = document.forms["RegForm"]["Name"];
                                var email = document.forms["RegForm"]["EMail"];
                                var phone = document.forms["RegForm"]["Telephone"];
                                var what = document.forms["RegForm"]["Subject"];
                                var password = document.forms["RegForm"]["Password"];
                                var address = document.forms["RegForm"]["Address"];


                                if (name.value == "") {
                                        window.alert("Please enter your name.");
```

```
                name.focus();
                return false;
            }
            if (address.value == "") {
                window.alert("Please enter your address.");
                address.focus();
                return false;
            }
            if (email.value == "") {
                window.alert(
                "Please enter a valid e-mail address.");
                email.focus();
                return false;
            }
            if (phone.value == "") {
                window.alert(
                "Please enter your telephone number.");
                phone.focus();
                return false;
            }
            if (password.value == "") {
                window.alert("Please enter your password");
                password.focus();
                return false;
            }
            if (what.selectedIndex < 1) {
                alert("Please enter your course.");
                what.focus();
                return false;
            }
            return true;
        }
    </script>
```

```
        <style>
            div {
                box-sizing: border-box;
                width: 100%;
                border: 100px solid black;
                float: left;
                align-content: center;
                align-items: center;
            }
            form {
                margin: 0 auto;
                width: 600px;
            }
        </style>
    </head>
    <body>
        <h1 style="text-align: center;">REGISTRATION FORM</h1>
        <form name="RegForm" action="/submit.php"
            onsubmit="return GEEKFORGEEKS()" method="post">
            <p>Name: <input type="text" size="65" name="Name" /></p> <br />
            <p>Address: <input type="text" size="65" name="Address"/></p><br />
            <p>E-mail Address: <input type="text" size="65" name="EMail" /></p>
            <br />
            <p>Password: <input type="text" size="65" name="Password" /></p>
            <br />
            <p>Telephone: <input type="text" size="65" name="Telephone" /></p>
            <br />
            <p>
                SELECT YOUR COURSE
                <select type="text" value="" name="Subject">
                    <option>BTECH</option>
                    <option>BBA</option>
                    <option>BCA</option>
                    <option>B.COM</option>
```

```
                    <option>GEEKFORGEEKS</option>
                </select>
            </p>
            <br />
            <br />
            <p>Comments: <textarea cols="55" name="Comment"> </textarea></p>
            <p>
                <input type="submit" value="send" name="Submit" />
                <input type="reset" value="Reset" name="Reset" />
            </p>
        </form>
    </body>
</html>
```

OUTPUT:

Resulting Form: