

ADVANCED JAVA(22ISL44) LAB PROGRAMS

LAB-01: Java program to demonstrate oracle database connectivity.

Step 1: Create table Emp in sql

```
CREATE TABLE Emp (EmpID int, FirstName varchar(255), LastName varchar(255));
```

Step 2: Insert few rows in to the table Employe

```
INSERT INTO Emp (EmpID, FirstName, LastName) VALUES (111, 'Abcd', 'Qwer');
```

```
INSERT INTO Emp (EmpID, FirstName, LastName) VALUES (112, 'Pqrs', 'Zxcv');
```

Step -3: Write a java program in Eclipse to connect to Oracle database

```
import java.sql.*;

class OracleCon{

    public static void main(String args[]){

        try{

            Class.forName("oracle.jdbc.driver.OracleDriver");

            Connection
            con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","admin");

            Statement stmt=con.createStatement();

            String str="select * from Emp";

            ResultSet rs=stmt.executeQuery(str);

            int empid = 0;

            String firstname, lastname;

            while(rs.next()) {

                empid = rs.getInt("EmpId");

                firstname = rs.getString("FirstName").trim();

                lastname = rs.getString("LastName").trim();
```

```

        System.out.println(empid + " " + firstname + " " + lastname );
    }

    con.close();

}

catch(Exception e){ System.out.println(e);}

}

}

```

LAB-02: Demonstration of http servlet.

getrequest.html →

```

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
Handling HTTP Get Request<br></br>
<form action='getReq'method='get'>
Enter your name
<input type='text' name='username' />
<br></br>
Enter your password:
<input type='password' name='password' />
<br></br>
<input type='submit' value='click me' />
</form>
</body>
</html>

```

MyServlet.java →

```

package second_prgm;

import javax.servlet.http.*;

import java.io.*;

import javax.servlet.*;

```

```

public class MyServlet extends HttpServlet
{
    protected void doGet(HttpServletRequest req,HttpServletResponse res)throws ServletException,
IOException{

        String un=req.getParameter("username");

        String pwd=req.getParameter("password");

        res.setContentType("text/html");

        PrintWriter pw=res.getWriter();

        pw.println("<h2>Welcome,"+un+"</h2>");

        pw.println("<br/><h2>your password is.."+pwd+"</h2>");

        pw.close();

    }

}

```

web.xml →

```

<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd" id="WebApp_ID" version="4.0">
    <display-name>second_prgm</display-name>
    <servlet>
        <servlet-name>GetReqServlet</servlet-name>
        <servlet-class>second_prgm.MyServlet</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>GetReqServlet</servlet-name>
        <url-pattern>/getReq</url-pattern>
    </servlet-mapping>
</web-app>

```

LAB-03: Write a JSP program to show the Fibonacci series up to a particular term, while the input is taken from an HTML form.

lab3.jsp→

```
<html>
<head>
<title> FIBONACCI SERIES IN JSP </title>
</head>
<body>
<form method="get">
<h3> Enter the number of terms you want:
<input type="text" name="limit">
</h3>
</form>
<h3>
<%
String s = request.getParameter("limit");
if (s != null) {
%>
<%@ page import = "java.io.*" %>
<%@ page import = "java.lang.*" %>
<%
int n=0;
n=Integer.parseInt(s);
out.println("No of terms to be printed is "+n);
%>
<br>
<br>
<br>
The series generated are listed below :<br><br>
<%
int a=1;
int b=1;
out.println(""+a+",\t"+b+",\t");
for(int i=3;i<= n;i++)
{
int c=a+b;
out.print(""+c+",\t");
a=b;
b=c;
}
}
%>
</h3>
</body>
</html>
```

LAB-04: Write a JSP program to calculate factorial values for a given integer number, take the input from a HTML form.

factorial.html →

```
<html>
<body>
<form action="factorial.jsp">
Enter a value for n: <input type="text" name="val">
<input type="submit" value="Submit">
</form>
</body>
</html>
```

factorial.jsp →

```
<html>
<body>
<%!
    long n,result;
    String str;

    long fact(long n) {
        if(n==0)
            return 1;
        else
            return n*fact(n-1);
    }
%>
<%
    str = request.getParameter("val");
    n=Long.parseLong(str);
    result = fact(n);
%>
<b>Factorial value:</b> <%= result %>
</body>
</html>
```

LAB-05: Write a JSP program to demonstrate the publisher database.

Before ,create a table named PUBLISH(in Run SQL) as follows:-

```
CREATE TABLE PUBLISH (  
    ID INT,  
    NAME VARCHAR(50),  
    CITY VARCHAR (25),  
    STATE VARCHAR (25),  
    COUNTRY VARCHAR (25)  
);  
  
INSERT INTO PUBLISH (ID, NAME, CITY, STATE, COUNTRY) VALUES (1, 'PHI', 'DELHI', 'DELHI', 'INDIA');
```

lab5.jsp →

```
<!DOCTYPE html>  
  
<HTML>  
  
<HEAD>  
    <TITLE>The Publishers Database Table </TITLE>  
</HEAD>  
<BODY>  
    <H1>The Publishers Database Table </H1>  
    <%@page contentType="text/html" pageEncoding="UTF-8"%>  
    <%@page import="java.sql.*" %>  
  
    <% Class.forName("oracle.jdbc.driver.OracleDriver");%>  
    <%  
  
    Connection con = null;  
  
    Statement statement= null;
```

```
ResultSet resultset = null;

con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","tiger");

statement = con.createStatement();

resultset = statement.executeQuery("select * from PUBLISH");

%>

<TABLE BORDER="1">

<TR>

<TH>ID</TH>

<TH>Name</TH>

<TH>City</TH>

<TH>State</TH>

<TH>Country</TH>

</TR>

<% while(resultset.next()){ %>

<TR>

<TD> <%= resultset.getString(1) %> </TD>

<TD> <%= resultset.getString(2) %> </TD>

<TD> <%= resultset.getString(3) %> </TD>

<TD> <%= resultset.getString(4) %> </TD>

<TD> <%= resultset.getString(5) %> </TD>

<% }

resultset.close();

statement.close();

con.close();

%>
```

```
</TR>
</TABLE>
</BODY>
</HTML>
```

LAB-06: Implement a JSP program to display Sine Table using java.lang.Math class.

lab6.jsp →

```
<html>
<head>
<title>sin theta</title>
<style>
table{
padding:6px;
th{
background-color:green;
padding:3px;
color:red;
}
</style>
</head>
<center>
<%!
    int degrees[]={0,15,30,45,60,75,90};

    double number;
```



```

String result;

%>

<table border="2" align="center">

<th>Degree</th><th>Sine Value</th>

<%

for (int i=0; i<degrees.length; i++) { //start for loop

    number = Math.sin(Math.toRadians(degrees[i]));

    result = String.format("%.2f", number);

%>

    <tr>

        <td><%= degrees[i] %></td>

        <td><%= result %></td>

    </tr>

<% } //end for loop %>

</table>

</center>

</body>

</html>

```

LAB-07:

A}. Write a JSP program to show the System date and time.

lab7a.jsp →

```

<html>

<head>

```

```

<%-- JSP comments --%>

<%@page import="java.util.Date"%>

<%!

    Date date;

%>

<%

    date = new Date();

%>

<b>System date and time: </b> <%= date %>

</body>

</html>

```

B} Write JSP program to display Stationary Order Form.

lab7b.jsp →

```

<HTML>

<HEAD>

    <TITLE>Stationary Order Form</TITLE>

</HEAD>

<BODY>

<H1 ALIGN="center"> Stationary Order Form</H1>

<%!

    String item[] = {"Pencil", "NoteBook", "Pen"};

    double price[] = {10.99, 35.99, 9.99};

    int quantity[] = {2, 9, 24};

%>

```

```

<TABLE ALIGN="center" BGCOLOR="lightgray" BORDER="1" WIDTH="75%">

<TR><TD>Item</TD>

<TD>Price</TD>

<TD>Quantity</TD>

<TD>Total Price</TD>

</TR>

<% for (int i=0; i<3; i++) { %>

    <TR><TD><%= item[i] %></TD>

    <TD><%= price[i] %></TD>

    <TD><%= quantity[i] %></TD>

    <TD><%= price[i] * quantity[i] %></TD>

    </TR>

<% } //end for loop %>

</TABLE>

</BODY>

</HTML>

```

LAB08: Implement a JSP program to demonstrate the use of JSP include action tag.

date.jsp →

```

<html>

<head>

<!-- JSP comments --%>

<%@page import="java.util.Date"%>

<%!

    Date date;

```

```
%>
```

```
<%
```

```
    date = new Date();
```

```
%>
```

```
<b>System date and time: </b> <%= date %>
```

```
</body>
```

```
</html>
```

```
orderform.jsp→
```

```
<HTML>
```

```
<HEAD>
```

```
    <TITLE>Stationary Order Form</TITLE>
```

```
</HEAD>
```

```
<BODY>
```

```
<H1 ALIGN="center"> Stationary Order Form</H1>
```

```
<%!
```

```
    String item[] = {"Pencil", "NoteBook", "Pen"};
```

```
    double price[] = {10.99, 35.99, 9.99};
```

```
    int quantity[] = {2, 9, 24};
```

```
%>
```

```
<TABLE ALIGN="center" BGCOLOR="lightgray" BORDER="1" WIDTH="75%">
```

```
<TR><TD>Item</TD>
```

```
<TD>Price</TD>
```

```
<TD>Quantity</TD>
```

```
<TD>Total Price</TD>
```

```
</TR>
```

```
<% for (int i=0; i<3; i++) { %>

    <TR><TD><%= item[i] %></TD>

    <TD><%= price[i] %></TD>

    <TD><%= quantity[i] %></TD>

    <TD><%= price[i] * quantity[i] %></TD>

    </TR>

<% } //end for loop %>
```

```
</TABLE>
```

```
</BODY>
```

```
</HTML>
```

```
action.jsp→
```

```
<html>
```

```
<head>
```

```
<title>JSP Page</title>
```

```
</head>
```

```
<body>
```

```
<font color="red">Include the First File:</font>
```

```
<br><br>
```

```
<jsp:include page="date.jsp"/>
```

```
<br><br><br>
```

```
<font color="blue">Include the Second File:</font>
```

```
<br>
```

```
<jsp:include page="orderform.jsp"/>
```

```
</body>
```

```
</html>
```

LAB-09: Program to implement session management in jsp.

Aim :-To implement session management in jsp.

Objective :-Students will able to perform session management in jsp.

FileName: **SessionObject.jsp**

```
<%@ page import="java.io.*, java.util.*"%>
<%
// Get session creation time.
Date createTime = new Date(session.getCreationTime());
// Get last access time of this Webpage.
Date lastAccessTime = new Date(session.getLastAccessedTime());
String title = "Welcome Back to my website";
Integer visitCount = new Integer(0);
String visitCountKey = new String("visitCount");
String userIDKey = new String("userID");
String userID = new String("USER123");
// Check if this is new comer on your Webpage.
if (session.isNew()) {
title = "Welcome to my website";
session.setAttribute(userIDKey, userID);
session.setAttribute(visitCountKey, visitCount);
}
visitCount = (Integer) session.getAttribute(visitCountKey);
visitCount = visitCount + 1;
userID = (String) session.getAttribute(userIDKey);
session.setAttribute(visitCountKey, visitCount);
%>
<html>
<head>
<title>Session Tracking</title>
</head>
<body>
<h1>Session Tracking</h1>
<table border="1">
<tr bgcolor="#949494">
<th>Session info</th>
<th>Value</th>
</tr>
<tr>
<td>id</td>
<td>
<%
out.print(session.getId());
%>
</td>
</tr>
<tr>
<td>Creation Time</td>
<td>
<%
out.print(createTime);
%>

```

```

</td>
</tr>
<tr>
<td>Time of Last Access</td>
<td>
<%
out.print(lastAccessTime);
%>
</td>
</tr>
<tr>
<td>User ID</td>
<td>
<%
out.print(userID);
%>
</td>
</tr>
<tr>
<td>Number of visits</td>
<td>
<%
out.print(visitCount);
%>
</td>
</tr>
</table>
</body>
</html>

```

LAB-10:Implementation of Remote method invocation(RMI).

client.java→

```
package package10;
```

```
import java.rmi.*;
```

```
import java.rmi.registry.*;
```

```
import java.rmi.Remote.*;
```

```
public class client {
```

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

```

        // TODO Auto-generated method stub

try
{
    String url="rmi://localhost:8080/calc";

    interface10 y=(interface10)Naming.lookup(url);

    System.out.println(y.add(20,10));

    System.out.println(y.sub(20,10));

    System.out.println(y.mul(20,10));

    System.out.println(y.div(20,10));

}

catch(Exception e)

{

    System.out.println("Failed");

}

}
}

```

Registration.java →

```

package package10;

import java.rmi.*;

import java.rmi.registry.*;

import java.rmi.Remote.*;

import java.rmi.registry.LocateRegistry;

public class Registration {

```



```

    public static void main(String[] args) {

        // TODO Auto-generated method stub

    try
    {

        server x=new server();

        System.out.println("Registering");

        Registry reg=LocateRegistry.createRegistry(8080);

        reg.rebind("calc", x);

        System.out.println("Registered");

    }

    catch(Exception e)

    {

        System.out.println("Registration Failed");

    }

    }

}

```

server.java→

```

package package10;

import java.rmi.RemoteException;

import java.rmi.registry.LocateRegistry;

import java.rmi.registry.Registry;

import java.rmi.server.UnicastRemoteObject;

```

```
public class server extends UnicastRemoteObject implements interface10{
```

```
    public server() throws RemoteException
```

```
    {
```

```
        super();
```

```
    }
```

```
    public int add(int a,int b)throws RemoteException
```

```
    {
```

```
        return a+b;
```

```
    }
```

```
    public int sub(int a,int b)throws RemoteException
```

```
    {
```

```
        return a-b;
```

```
    }
```

```
    public int mul(int a,int b)throws RemoteException
```

```
    {
```

```
        return a*b;
```

```
    }
```

```
    public int div(int a,int b)throws RemoteException
```

```
    {
```

```
        return a/b;
```

```
    }
```

```
}
```

interface10.java→

```
package package10;
```

```
import java.rmi.*;
```

```
import java.rmi.registry.*;
```

```
import java.rmi.Remote.*;
```

```
public interface interface10 extends Remote {
```

```
    public int add(int a,int b)throws RemoteException;
```

```
    public int sub(int a,int b)throws RemoteException;
```

```
    public int mul(int a,int b)throws RemoteException;
```

```
    public int div(int a,int b)throws RemoteException;
```

```
}
```